Escola de Camins
Escola Tècnica Superior d'Enginyeria de Camins, Canals i Ports
UPC BARCELONATECH

TREBALL FINAL DE MÀSTER

Sensitivity analysis and experimental calibration for the Additive Manufacturing computational framework.

Treball realitzat per:
Tomás Varona Poncela

Dirigit per:
Michele Chiumenti & Miguel Cervera

Màster en:
Numerical Methods in Engineering

Barcelona, 22 de Junio de 2016

Departament of Civil and Environmental Engineering

POLYTECHNIC UNIVERSITY OF CATALONIA

MASTER THESIS

Sensitivity analysis and experimental calibration for the Additive Manufacturing computational framework

Author:
Tomás VARONA

Advisers:
Prof. Michele CHIUMENTI
Prof. Miguel CERVERA

Master on Numerical Methods in Engineering

Barcelona, June 22, 2016

Escola de Camins
Escola Tècnica Superior d'Enginyeria de Camins, Canals i Ports
UPC BARCELONATECH

"Make things as simple as possible, but not simpler."

Albert Einstein

Abstract

A reliable computational framework for the simulation of additive manufacturing by blown powder is crucial for its profitable use in the industrial world. Therefore, this Master Thesis addresses a validation of an in-house finite element software comparing numerical results with campaign data. This data is obtained at SKLSP laboratories in China where a Laser Solid Forming machine is employed to fabricate metal parts directly from CAD models. Additionally, a sensitivity analysis is performed in order to acquire a better knowledge about the influence of material properties on the results.

The numerical simulation of additive manufacturing involves the interaction of thermal, mechanical and metallurgical phenomena. For this reason, a large number of material properties are studied in this work. An exhaustive set of simulations has been performed varying several mechanical parameters, such as Poisson's ratio, Young's modulus or viscosity, and studying the response in terms of vertical displacements. In an orderly way, the results were stored and plotted to, subsequently, be compared with the experimental data in the calibration and with each others in the sensitivity analysis.

The experimental calibration has demonstrated a notorious agreement with the campaign evidence after two major changes: suitable use of Poisson's ratio temperature dependent values and the introduction of a visco-elastic material response in the constitutive model. The sensitivity analysis has provided a deeper understanding of the phenomenon, placing plasticity, visco-elasticity, and phase change as the main mechanisms which influence the simulation.

KEYWORDS: Additive manufacturing (AM), blown powder technology, Laser Solid Forming (LSF), FE model, thermo-mechanical analysis

Acknowledgements

I would like to thank all the people who have contributed to make possible the fulfillment of this Master Thesis. First and foremost, I would like to express my sincere gratitude to my academic advisers, Prof. Michele Chiumenti and Prof. Miguel Cervera, for accepting me into their group believing in me from the beginning. Their advice, patience, motivation and wide knowledge have been crucial during this dissertation helping to improve my scarcities and strengthen my virtues.

Besides my advisers, I would like to acknowledge the International Center for Numerical Methods in Engineering (CIMNE) for giving me the opportunity to be a member of their staff. Also the financial support from the CA*xMan* Project (Project Reference 680448)- *Computer Aided Technologies for Additive Manufacturing* - funded under the *Horizon 2020* EU Framework Programme (H2020-EU.2.1.1., H2020-EU.2.1.5., H2020-EU.2.1.5.1) is gratefully acknowledged.

I sincerely thank the COMET team (COupled MEchanical and Thermal Analysis) for providing the code employed during this Master Thesis and the *State Key Laboratory of Solidification Processing* (SKLSP) in China for measuring the experimental data which is used to calibrate the numerical model.

I would also like to thank my parent who gave me their unconditional support, my Santander and Barcelona friends who make the best of them to cheer me up wherever I am, and my colleagues for enlivening the long working hours during these months.

Last but not the least, I would like to express my gratitude to Eric who is both a colleague and a good friend. His advices are undoubtedly invaluable and, indeed, have helped to lead my steps to the place I am.

Contents

List of Figures

List of Tables

List of Abbreviations

AM	**A**dditive **M**anufacturing
LENS	**L**aser **E**ngineered **N**et **S**haping
COMET	**CO**upled **ME**chanical **T**hermal
LSF	**L**aser **S**olid **F**orming
CAD	**C**omputer **A**ided **D**esign
HTC	**H**eat **T**ransfer **C**oefficient
HAZ	**H**eat **A**ffected **Z**one
CLI	**C**ommon **L**ayer **I**nterface
SKLSP	**S**tate **K**ey **L**aboratory **S**olidification **P**rocessing
VD	**V**ertical **D**isplacement

Physical Constants

Stefan-Boltzmann $\sigma_{rad} = 5.670373 \cdot 10^{-8} \frac{W}{m^2 \cdot K^4}$

List of Symbols

H	enthalpy	J
Q	heat source	J
D_{mech}	thermomechanical dissipation	J
$\mathbf{q}$	heat flux	Wm^{-2}
k	thermal conductivity	$Wm^{-1}K^{-1}$
T	temperature	K
C	heat capacity	$Jm^{-3}K^{-1}$
L	latent heat	Jm^{-3}
f_S	solid fraction	
f_L	liquid fraction	
h	heat transfer coefficient	$Wm^{-2}K^{-1}$
$\mathbf{u}$	displacement field	m
$\mathbf{b}$	body forces	Nm^{-2}
p	hydrostatic pressure	Nm^{-2}
$\mathbf{s}$	deviatoric stress tensor	Nm^{-2}
e^T	thermal deformation	
K	bulk modulus	Nm^{-2}
e^{vol}	volumetric part strain tensor	
$\mathbf{e}$	deviatoric part strain tensor	
e^{cool}	thermal contraction strain	
e^{pc}	thermal shrinkage strain	
R	yield surface radius	
q	stress-like variable	Nm^{-2}
P	total laser energy	J
v_{MD}	scanning speed	ms^{-1}
v_{back}	laser back-speed	ms^{-1}
E	Young modulus	Nm^{-2}
G	shear modulus	Nm^{-2}
ε_{rad}	radiation emissivity	
$\boldsymbol{\sigma}$	Cauchy stress tensor	Nm^{-2}
Ω	Volume domain	
Γ_u	Dirichlet boundary	
$\Gamma_{\boldsymbol{\sigma}}$	Neumann boundary	
$\boldsymbol{\varepsilon}$	Total strain tensor	
α	thermal expansion	K^{-1}
$\mathbf{\Phi}$	yield surface	
σ_y	yield stress	Nm^{-2}
σ_∞	Saturation flow stress	Nm^{-2}
ξ	strain-hardening variable	
$\boldsymbol{\varepsilon^{vp}}$	visco-plastic strain tensor	
γ^{vp}	visco-plastic multiplier	
η	viscosity parameter	Nsm^{-2}
η_p	heat absorption coefficient	

τ	retardation time	s
β	visco-elastic infinite ratio	
ν	Poisson ratio	

Chapter 1

Introduction

1.1 Concept and history of additive manufacturing

Additive manufacturing technology (AM) consists of building parts, layer upon layer, directly from 3D model data. The Computer-Aided Design (CAD) is transformed to surface mesh (STL format) which thanks to a slicing tool is divided in the required layers. This information is employed by the 3D printer to build the part. The work-flow of additive manufacturing can be observed in FIGURE 1.1[1] [2].

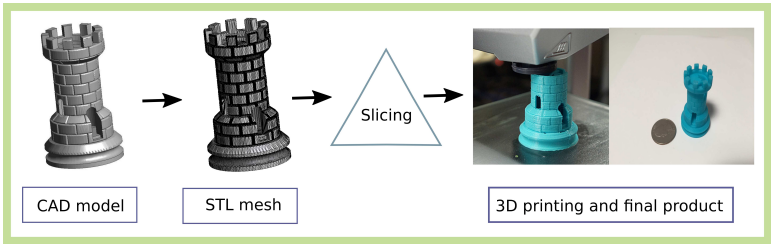


FIGURE 1.1: Additive manufacturing work-flow

This technology is far to be considered novel. It was born thirty years ago under the name of rapid prototyping [6]. This term emphasized the idea of quickly basis model creation from which the final product can be derived. However, nowadays, due to the improvement in computer technology (processing power, graphics capability, machine control), the increase of material properties knowledge, and the development of manufacturing process quality, restricting AM possibilities to its original conception is unthinkable, that is, AM has evolved to a legitimate manufacturing process [7, 25].

During its history, a great number of printing processes have been invented, such as stereolithography [4], selective laser sintering (powder bed), wire-feeding or laser engineered net shaping (LENS) [30]. The aforesaid technologies are shown in FIGURE 1.2[2]. In this dissertation, the technique to be experimentally calibrated is laser engineered net shaping (blown powder), consisting of melting metal powder which is injected into a specific

[1] Images:http://www.pcmag.com/article2/0,2817,2418103,00.asp,https://www.3dhubs.com/los-angeles/hubs/rusty

[2] Images:https://www.stratasysdirect.com/blog/how-it-works-stereolithography/,https://3dprint.com/13367/astm-international-standards/,http://www.machinery.co.uk/machinery-features/additive-manufacturing-impact-on-subtractive-machining-found-to-be-minimal/118892/,http://www.sciaky.com/additive-manufacturing/electron-beam-additive-manufacturing-technology

location thanks to the actuation of a high-powered laser beam [28].

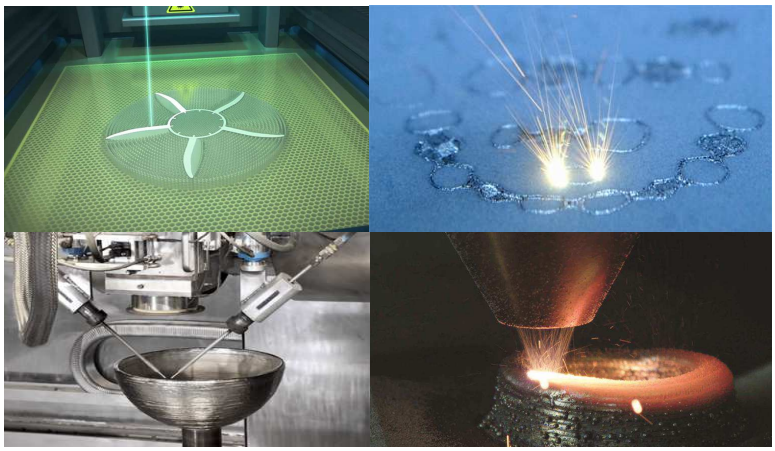


FIGURE 1.2: Additive manufacturing technologies. From up to down and left to right: stereolithography, selective laser sintering, wire-feeding, and laser engineered net shaping

1.2 Strengths and weaknesses of additive manufacturing

Attending to the wide range of possibilities provided by the AM techniques, the positioning of AM as the cornerstone of a so-called third industrial revolution is not surprising[3]. Some of these remarkable advantages are [7, 25]:

- *Freedom of design*: allowing building parts with very complex shapes which were expensive or unfeasible to produce with old-fashioned techniques.
- *Seamlessness*: reducing the process steps.
- *Reduce costs*: material optimization, avoid moulds or hand carving.
- *High accuracy*: due to laser thin spot, the accuracy could be up to tens of microns.
- *Small lot production*: the lack of moulds can lead to a profitable production of short number of units.

Contrariwise, AM suffers from some drawbacks which could complicate its industrial use. The freedom of design leads to intricate shapes which can be built by numerous combinations of the process parameters which contributes to an uncertainty when trying to design the geometry, choose the material, and customize the AM procedure.

[3] "The third industrial revolution" - The Economist

1.3 Challenges of the numerical simulation

This work concerns the numerical simulation of additive manufacturing (3D printing) by blown powder. The correct achievement of an accurate and reliable simulation must take into account two main aspects. First of all, a coupled thermo-mechanical scheme is to be considered in order to capture the real process. Secondly, the constitutive model for the material is complex taking into account plasticity and viscous behaviour. These two hurdles lead to the necessity of including a wide range of material and process parameters which are mainly temperature dependent. Some of them are known "a priori", however, others, such as high temperature Poisson's ratio or linear isotropic hardening, are difficult to measure. From a point of view of accurately simulating additive manufacturing is important to have a correct calibration of these parameters for each target material. Additionally, the knowledge of the influence in the variation of these parameters is crucial for material science and process optimization. The accurate simulation of AM processes will avoid the need for trial and error printing parts until finding the suitable parameters configuration. This trial and error is linked to an increase in the production costs and, in some cases, the inability to undertake certain kind of businesses.

1.4 Key participants of the work

The finite element code employed in this Master Thesis has been developed by the COMET team (COupled MEchanical and Thermal Analysis) [14]. The supervisors of this dissertation, Prof. Michele Chiumenti and Prof. Miguel Cervera, are both main developers of COMET, permanent staff of CIMNE (International Center for Numerical Methods in Engineering), and Professors at the Technical University of Catalonia. Their career path in the field of numerical methods and, specifically, solving thermo-mechanical problems applied to metal deposition and working with different constitutive models is remarkable. Therefore, an extensive, powerful, and rich literature support (Chapter 2), as well as, a robust and reliable finite element code is provided by the supervisors for the correct completion of this dissertation. It has been crucial the collaboration of *State Key Laboratory of Solidification Processing* (SKLSP) in China for measuring the experimental data which is used to calibrate the numerical model.

1.5 Objectives of the thesis

Taking into account the large number of parameters involved in the process and the unlimited capabilities of having an accurate simulation of AM, the importance of performing a sensitivity analysis and an experimental calibration of the AM computational framework, which is the goal of this Master Thesis, is irrefutable. The dissertation is divided into two objectives: the sensitivity analysis of material parameters and the experimental calibration comparing the numerical results with the experimental campaign carried out at SKLSP laboratories, where a Laser Solid Forming technique (LSF) [29], or Laser Engineered Net Shaping (LENS), is employed to fabricate

metal parts directly from CAD models. For this comparison, the pre and postprocessor GiD [23] has been employed in order to construct the geometry, impose boundary conditions and mesh, and compute the results afterwards. The thermal experimental calibration of the computational framework for AM by blown powder is already fulfilled matching successfully the experimental campaign [18]. Therefore, the starting point is promising and the inclusion of the mechanical part in the analysis is performed on the basis of a correct thermal calibration.

1.6 Summary of methodology

In chapter 4, the methodology of this Master thesis is presented. It is divided in two main parts: the experimental calibration and the sensitivity analysis. Hereafter, a brief summary about the information which can be found within this methodology is provided.

1.6.1 Methodology: experimental calibration

The experimental calibration consists of comparing the campaign data collected in SKLSP laboratories in China with the numerical simulations. In this section, the following points are tackled:

- *Experimental campaign*: extended information about the experimental campaign paying attention to the instrumental, the machine, the samples, the measurement gauges, among others, is performed. Additionally, a full machine data with the main welding parameters is also provided.
- *Computational model*: The numerical model employed to compute the simulation is explained in this point. A detailed information about geometry and mesh, boundary conditions, and material characterization is included.
- *Simplified model*: A computationally cheap model is employed in order to rapidly run simulations varying target parameters. A set of python subroutines to automatize this simulations is also detailed in this point.
- *Visco-elastic scheme*: a visco-elastic scheme is implemented in order to be able to calibrate the cooling phase of the AM.

1.6.2 Methodology: sensitivity analysis

The sensitivity analysis intends to provide a deeper understanding of the influence which the material parameters have in the simulation. The features which are studied in this sensitivity analysis are:

- *Sensitivity to plasticity*: the different tests which are run in order to capture the effect of plastification are explained in this point.

- *Sensitivity to visco-elasticity*: the implementation, which has been already done, depends on several parameters which are mentioned in this point.
- *Sensitivity to phase change*: the procedure followed in order to assess the influence of the phase change in the mechanical result is included in this point.

1.7 Contents

The document is organized following an usual scientific dissertation structure. Before starting the chapters with the introduction, the following parts are included:

- *Title page*
- *Abstract and keywords*
- *Acknowledgements*
- *List of: contents, figures, tables, abbreviations, symbols, and physical constants*

Afterwards, the main chapters are included. This Master thesis is divided into six chapters which are:

- *Chapter 1- Introduction*
- *Chapter 2- Literature review*
- *Chapter 3- State of the art*
- *Chapter 4- Methodology*
- *Chapter 5- Results and discussion*
- *Chapter 6- Conclusions*

Finally, to close the dissertation, the bibliography and the appendices are presented. There are three appendices in which the following information is included:

- *Material specifications*
- *Welding specifications*
- *Python code*

Chapter 2

Literature review

In this chapter, a more detailed explanation about the bibliography employed is provided. The main references, which have been extensively used during this dissertation, are those written by the thesis advisers: Prof. Michele Chiumenti and Prof. Miguel Cervera, and some researches from CIMNE (International Center for Numerical Methods in Engineering). These references, which are the base of the state of the art, the finite element code and the current *know-how* of additive manufacturing are: [3, 4, 14, 15, 17, 18, 26]. Additionally, some other references in the world of additive manufacturing are employed such as: [1, 2, 6, 7, 8, 13, 19, 22, 25, 28, 29, 30]. From this references, the extracted information reads about the social impact of additive manufacturing, industrial applications, different AM techniques, solidification processes, among others.

Apart from the specific bibliography on additive manufacturing, some additional research was done in order to obtain a deeper knowledge in the following features:

- Gid, the personal pre and post processor: [23].
- Inelastic scheme to implement the visco-elasticity material behaviour: [11].
- Specific numerical methods references: [16, 21].
- References in Python language: [5, 9]
- References for plotting the images with gnuplot: [27]

Finally, during all the dissertation process, a book on Master Thesis writing [20] and other about LaTeX [12] have been constantly consulted.

Chapter 3

State-Of-The-Art

In this chapter, the computational framework employed to simulate the blown powder additive manufacturing (AM), as well as, a deeper explanation of the specific AM technique is presented. The main aspects of the computational framework which are to be explained are: heat transfer analysis, the stress analysis, the material constitutive model, the finite element modeling, and the definition of scanning-path.

3.1 Additive manufacturing technique: Laser Engineered Net Shaping

Laser Engineered Net Shaping (LENS) [18] is included in the family of AM processes called blown powder or powder feeding. In this process, the powder is injected co-axially by a feed nozzle directly into a laser beam which melts the particles on a substrate to form a metallurgical bond when cooled. The substrate is, when some layers have been deposited, the previous ones and, therefore, a penetration must be taken into account. In FIGURE 3.1, a schematic representation of the process can be observed, the melting pool is created within the previous deposited layers, it contributes to a better material sintering.

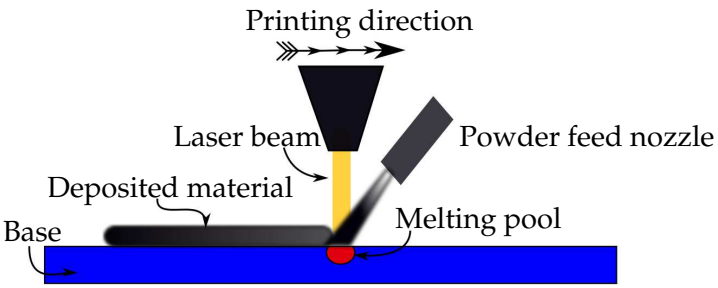


FIGURE 3.1: LENS process

The generated deposit [22] may range from 0.127 to 1.016 mm thick and 1.016 to 4.064 mm wide. Motion control for the deposit can be programmed manually or may be generated from CAD files processed directly by the system's software. They are typically performed in a controlled argon atmosphere containing less than 10 ppm oxygen. Due to the small melting pool and high travel speeds, the deposits cool very fast (up to 10 $\frac{^oC}{s}$), leading to fine grain structures.

The high quality of the deposits is the strong point of this technology. It is the reason why it is being evaluated in the medical industry, aerospace,

and Department of Defense, as well as, commercial industries that include electric power generation, oil/gas, chemical processing, and mining. The versatility and flexibility of the process is observed in applications where a variety of materials are deposited on different geometries. The savings in cost, time and material are impressive [22].

3.2 Computational framework

The computational framework consists of a finite-element based code [14] written in Fortran which solves the coupled thermo-mechanical [3] problem, adding certain particular features for taking into account the blown powder additive manufacturing technology. An explanation of the thermo-mechanical (heat transfer [24] and stress analysis), constitutive model and finite element theories and hypothesis is crucial in order to understand the capabilities and limits of the code[1].

3.2.1 Heat transfer analysis

The additive manufacturing is performed in two stages: printing and cooling. Both thermal phases are governed by the *energy balance equation* which can be written as [18]:

$$\dot{H} = -\nabla \cdot \mathbf{q} + \dot{Q} + \dot{D}_{mech} \tag{3.1}$$

where $\dot{H}$, $\dot{Q}$, and $\dot{D}_{mech}$ represent the enthalpy, heat source, and thermo-mechanical dissipation rates (per unit of volumen) respectively, and $\mathbf{q}$ is the heat flux (per unit of surface) which could be expressed as a function of the temperature gradient through Fourier's law as:

$$\mathbf{q} = -k\nabla T \tag{3.2}$$

being $k(T)$ the temperature dependent thermal conductivity. Taking into account the high conductivity of metals, the thermal diffusion process can be considered the key mechanism of heat transfer in metal deposition. The heat source is the energy input introduced into the melting pool along the scanning path. Comparing with this power source of the laser beam, the thermo-mechanical dissipation could be neglected.

Before integrating the energy balance equation (3.1),the enthalpy state variable, H, is to be expressed as a function of the temperature, T, and the liquid fraction, f_L, and, therefore, the enthalpy rate can be written as:

$$\dot{H}(T, f_L) = \frac{\partial H}{\partial T}\dot{T} + \frac{\partial H}{\partial f_L}\dot{f}_L = C\dot{T} + L\dot{f}_L \tag{3.3}$$

being $C(T) = \frac{\partial H}{\partial T}$ the temperature dependent heat capacity and $L(T) = \frac{\partial H}{\partial f_L}$ the latent heat released in the phase-change process. Once the terms are defined, the weak form of the problem can be stated.

[1] Code implementation details are out of the scope of this dissertation [19, 21]

Let Ω be an open, bounded domain (Figure 3.2) in $\mathbb{R}^n$ where n is the number of dimensions of the space, closed by the smooth boundary $\Gamma = \Gamma_D \cup \Gamma_N$ in which the essential or Dirichlet (temperature) and natural or Neumann (heat flux) boundary conditions are imposed respectively.

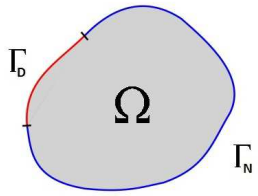


FIGURE 3.2: Continuum domain

$$T = T_D \qquad on \quad \Gamma_D \tag{3.4}$$

$$\mathbf{q} \cdot \mathbf{n} = q_n \qquad on \quad \Gamma_N \tag{3.5}$$

Multiplying the energy balance equation (3.1) by a test function expressed in terms of the variation of the temperature field (δT) and integrating over the domain Ω, it can be obtained the following integral expression:

$$\int_\Omega \Big[\dot{H}\delta T\Big] d\Omega = -\int_\Omega \Big[(\nabla \cdot \mathbf{q})\delta T\Big] d\Omega + \int_\Omega \Big[(\dot{Q} + \dot{D}_{mech})\delta T\Big] d\Omega \tag{3.6}$$

Applying the derivation chain rule to the first integral of the right hand side, this term can be split in a more suitable way for our purpose.

$$(\nabla \cdot \mathbf{q})\delta T = \nabla \cdot (\mathbf{q}\delta T) - \mathbf{q} \cdot \nabla(\delta T) \tag{3.7}$$

Subtituting this expression (3.7) into (3.6), appliying Fourier's law (3.2) and gathering the terms involving the unknown variable (temperature) in the left hand side, it reads:

$$\int_\Omega \Big[(C\dot{T} + L\dot{f}_L)\delta T\Big] d\Omega + \int_\Omega \Big[k\nabla T \cdot \nabla(\delta T)\Big] d\Omega = W^{ext}_{ther} \qquad \forall \delta T \tag{3.8}$$

the right hand side W^{ext}_{ther} denotes the external work of the thermal loads and it can be written as:

$$W^{ext}_{ther} = -\int_\Omega \Big[\nabla \cdot (\mathbf{q}\delta T)\Big] d\Omega + \int_\Omega \Big[(\dot{Q} + \dot{D}_{mech})\delta T\Big] d\Omega \tag{3.9}$$

the divergence theorem can be applied to the first integral in W^{ext}_{ther} in the following way:

$$\int_\Omega \Big[\nabla \cdot (\mathbf{q}\delta T)\Big] d\Omega = \int_\Gamma \Big[(\mathbf{q} \cdot \mathbf{n})\delta T\Big] d\Gamma \tag{3.10}$$

the smooth boundary Γ is defined as $\Gamma = \Gamma_D \cup \Gamma_N$ and the test function (δT) vanishes in the Dirichlet boundary, therefore, just the Neumann

boundary remains and $\mathbf{q} \cdot \mathbf{n}$ represent the prescribed normal fluxes (q_n), finally reading:

$$W_{ther}^{ext} = \int_{\Omega} \Big[(\dot{Q} + \dot{D}_{mech}) \delta T \Big] d\Omega - \int_{\Gamma_N} \Big[q_n \delta T \Big] d\Gamma \tag{3.11}$$

substituting the previous expression into (3.8), the resulting weak form of the energy balance equation is obtained.

The term q_n can be split into four heat flux mechanisms: prescribed heat flux ($\bar{q}$), heat loss by convection (q_{conv}), heat flux by conduction (q_{cond}), and the radiation heat flux (q_{rad}). The normal flux can, therefore, be written as follows:

$$q_n = \bar{q} + q_{conv} + q_{cond} + q_{rad} \tag{3.12}$$

The prescribed heat flux, $\bar{q}$, defined in EQUATION 3.12, can be used to spread part of power input over the surface of the substrate.

The heat loss by convection, q_{conv}, can be computed by means of Newton's law as:

$$q_{conv} = h_{conv}(T - T_{env}) \tag{3.13}$$

where h_{conv} is the the temperature dependent heat transfer coefficient (HTC) by convection, T is the material surface temperature, and T_{env} the temperature of the surrounding environment. It is common the assumption of a constant environment temperature during the additive manufacturing. However, if the process is carried out in close chamber with controlled atmosphere, a noticeable amount of energy scattered in it leading to an increment of the temperature which can not be neglected.

Similarly, the heat flux produced by the thermal conduction phenomenon between the component and the clamping system is expressed as follows:

$$q_{cond} = h_{cond}(T - T_{clamp}) \tag{3.14}$$

where h_{cond} is the temperature dependent heat transfer coefficient (HTC) by conduction, T the material surface temperature, and T_{clamp}. This HTC by conduction is defined as the inverse of the thermal resistivity between the materials in contact.

Finally, the main dissipation mechanism is the radiation heat flux, q_{rad}, due to the high temperature field induced by the heat source. This radiation heat flux can be computed by means of Stefan-Boltzmann's law as:

$$q_{rad} = \sigma_{rad} \varepsilon_{rad} (T^4 - T_{env}^4) \tag{3.15}$$

where σ_{rad} is the Stefan-Boltzmann constant and ε_{rad} the emissivity parameter. This equation can be rewritten in order to follow the same structure than EQUATIONS 3.13 and 3.14:

$$q_{rad} = h_{rad}(T - T_{env}) \tag{3.16}$$

where h_{rad} is the temperature dependent heat transfer coefficient (HTC) by radiation defined as:

$$h_{rad}(T) = \sigma_{rad}\varepsilon_{rad}(T^3 + T^2T_{env} - TT_{env}^2 - T_{env}^3) \tag{3.17}$$

This format is suitable for two reasons: first, it is possible to linearize the contribution of the heat radiation term. Second, it is extremely difficult to separate the heat losses due to convection and radiation and, therefore, the numerical model can assume a unique HTC as:

$$q_{loss} = h_{loss}(T - T_{env}) \tag{3.18}$$

where h_{loss} is the temperature dependent heat transfer coefficient accounting the total heat loss through the surrounding environment.

3.2.2 Stress analysis

The mechanical problem is solved by integrating the *momentum balance equation* [16, 17]. In local form, considering quasi-static conditions, the problem can be stated as: find the displacement field $\mathbf{u}$, for given prescribed forces $\mathbf{b}$, such that:

$$\nabla \cdot \boldsymbol{\sigma}(\mathbf{u}) + \mathbf{b} = \mathbf{0} \tag{3.19}$$

where $\sigma(\mathbf{u})$ denotes the Cauchy stress tensor. This stress tensor can be divided into its hydrostatic (pressure) and deviatoric parts:

$$\boldsymbol{\sigma}(\mathbf{u}, p) = p\mathbf{I} + \mathbf{s}(\mathbf{u}) \tag{3.20}$$

being $p = \frac{1}{3}tr(\boldsymbol{\sigma})$ and $\mathbf{s} = dev(\boldsymbol{\sigma})$. This split is very convenient when dealing with isochoric behaviour where deformations are mainly deviatoric (liquid-phase). Therefore, momentum balance equation (3.19) can be rewritten in the following way:

$$\nabla \cdot \mathbf{s}(\mathbf{u}) + \nabla p + \mathbf{b} = \mathbf{0} \tag{3.21}$$

$$(\nabla \cdot \mathbf{u} - e^T) - \frac{p}{K} = 0 \tag{3.22}$$

where $e^T$is the thermal deformation and $K(T)$ is the temperature dependent bulk modulus. This formulation is known as mixed $\mathbf{u}/p$ formulation and in suitable for compressible and incompressible analysis. When the material reaches the liquidus state, the thermal deformation is neglected ($e^T \approx 0$) and the bulk modulus tends to infinity ($K \to \infty$), so equation (3.22) enforces the volumetric constraint as:

$$\nabla \cdot \mathbf{u} = 0 \tag{3.23}$$

Similarly to the previous section for the heat transfer analysis, the split equations are to be integrated over the volume domain Ω being Γ the corresponding boundary $\Gamma = \Gamma_{\mathbf{u}} \cup \Gamma_{\boldsymbol{\sigma}}$ where $\Gamma_{\mathbf{u}}$ is the Dirichlet boundary (imposed displacements) and $\Gamma_{\boldsymbol{\sigma}}$ is the Neumann boundary (prescribed tractions). The weak integral form of the mechanical problem defined by the mixed $\mathbf{u}/p$ can be written as follows:

$$\int_\Omega \Big[(\nabla \cdot \mathbf{s}(\mathbf{u})) \cdot \delta\mathbf{u}\Big] d\Omega + \int_\Omega (\nabla p \cdot \delta\mathbf{u}) d\Omega + \int_\Omega (\mathbf{b} \cdot \delta\mathbf{u}) d\Omega = 0 \quad \forall \delta\mathbf{u} \quad (3.24)$$

$$\int_\Omega \Big[(\nabla \cdot \mathbf{u} - e^T - \frac{p}{K})\delta p\Big] d\Omega = 0 \quad \forall \delta p \quad (3.25)$$

where $\delta\mathbf{u}$ and δp are the variations of displacements and pressure field, respectively. If integration by parts is applied to first two terms of equation (3.24), the following expressions are obtained:

$$\int_\Omega \Big[(\nabla \cdot \mathbf{s}(\mathbf{u})) \cdot \delta\mathbf{u}\Big] d\Omega = -\int_\Omega (\mathbf{s}(\mathbf{u}) : \nabla^s \delta\mathbf{u}) d\Omega + \int_{\Gamma_{\boldsymbol{\sigma}}} (\bar{\mathbf{t}} \cdot \delta\mathbf{u}) d\Gamma \quad (3.26)$$

$$\int_\Omega (\nabla p \cdot \delta\mathbf{u}) d\Omega = -\int_\Omega (p \nabla \cdot \delta\mathbf{u}) d\Omega \quad (3.27)$$

being $\bar{\mathbf{t}}$ the prescribed tractions on $\Gamma_{\boldsymbol{\sigma}}$. Substituting equations (3.26) and (3.27) into (3.24), the mixed $\mathbf{u}/p$ variational form of the quasi-static mechanical problem is:

$$\int_\Omega (\mathbf{s}(\mathbf{u}) : \nabla^s \delta\mathbf{u}) d\Omega + \int_\Omega (p \nabla \cdot \delta\mathbf{u}) d\Omega = W^{ext}_{mech} \quad \forall \delta\mathbf{u} \quad (3.28)$$

$$\int_\Omega \Big[(\nabla \cdot \mathbf{u} - e^T - \frac{p}{K})\delta p\Big] d\Omega = 0 \quad \forall \delta p \quad (3.29)$$

where W^{ext}_{mech} denotes the external work of the mechanical loads defined as:

$$W^{ext}_{mech}(\delta\mathbf{u}) = \int_\Omega (\mathbf{b} \cdot \delta\mathbf{u}) d\Omega + \int_{\Gamma_{\boldsymbol{\sigma}}} (\bar{\mathbf{t}} \cdot \delta\mathbf{u}) d\Gamma \quad (3.30)$$

3.2.3 Constitutive model

The material model employed to simulate the constitutive behaviour should be chosen taking into account the wide range of temperatures experienced during the process [17]. The material response must reproduce both the elasto-plastic behaviour at room temperature and the pure viscous one observed above the melting point. This transition can be simulated adopting an *apropos* thermo-elasto-visco-plastic [11, 26] constitutive model. Reaching the melting point, the viscous behaviour becomes predominant, the elastic limit gradually vanishes. Consequently, the purely viscous model is recovered for the liquid-like behaviour. The constitutive laws for all the temperature range of the process are characterize by a J2-thermo-elasto-visco-plastic constitutive model of the form collected in EQUATIONS 3.31 and 3.32.

$$p = \frac{K}{f_S(T)}(e^{vol} - e^T) \quad (3.31)$$

$$\mathbf{s} = \frac{G}{f_S(T)}(\mathbf{e} - \mathbf{e}^{vp}) \quad (3.32)$$

where $K(T)$ and $G(T)$ are the bulk and shear moduli respectively. The total strain tensor can be expressed in terms of the displacement field, $\mathbf{u}$, as reads in EQUATION 3.33.

$$\boldsymbol{\varepsilon}(\mathbf{u}) = \nabla^S \mathbf{u} \tag{3.33}$$

and its volumetric and deviatoric parts are obtained as follows in EQUATIONS 3.34 and 3.35 respectively.

$$e^{vol} = tr(\boldsymbol{\varepsilon}) = \nabla \cdot \mathbf{u} \tag{3.34}$$

$$\mathbf{e} = dev(\boldsymbol{\varepsilon}) = \boldsymbol{\varepsilon} - \frac{e^{vol}}{3}\mathbf{I} \tag{3.35}$$

The thermal deformation which appears in EQUATION 3.31 is defined as a volumetric component of form collected in EQUATION 3.36.

$$e^T(T) = e^{cool}(T) + e^{pc} \tag{3.36}$$

where e^{pc} is the thermal shrinkage characteristic of the phase transformation and e^{cool} the thermal contraction developed during the cooling process from solidus to room temperature. The thermal shrinkage can be defined attending the principle of mass conservation as it is shown in EQUATION 3.37 in terms of the volume shrinkage, ΔV^{pc}.

$$e^{pc}(T) = \frac{\Delta V^{pc}}{V_0} = \frac{\rho(T) - \rho_L}{\rho_S} \qquad \rho_L \leq \rho(T) \leq \rho_S \tag{3.37}$$

where $\rho_S = \rho(T_S)$ and $\rho_L = \rho(T_L)$ are the material densities at solidus and liquidus temperatures. On the other hand, the thermal contraction, e^{cool}, during the cooling phase is written in EQUATION 3.38.

$$e^{cool}(T) = 3[\alpha(T)(T - T_{env}) - \alpha(T_0)(T_0 - T_{env})] \tag{3.38}$$

where $\alpha(T)$ is the temperature dependent secant thermal expansion coefficient, and T_0 is the initial temperature.

The J2-yield surface definition, $\Phi(\mathbf{s}, q, T)$ is presented in EQUATION 3.39.

$$\Phi(\mathbf{s}, q, T) = \|\mathbf{s}\| - f_S(T)R(q, T) \leq 0 \tag{3.39}$$

where $R(q, T)$ is the temperature dependent yield-surface radius defined as written in EQUATION 3.40.

$$R(q, T) = \sqrt{\frac{2}{3}}[\sigma_y(T) - q] \tag{3.40}$$

where $\sigma_y(T)$ is the temperature dependent initial yield stress (elastic limit) parameter. The stress-like variable, $q(\xi, T)$, conjugate to the isotropic strain-hardening variable, ξ, controls the isotropic hardining effect defined in EQUATION 3.41.

$$q(\xi, T) = -[\sigma_\infty(T) - \sigma_y(T)]\left[1 - e^{-\delta(T)\xi}\right] - h(T)\xi \tag{3.41}$$

being $h(T)$ and $\delta(T)$ are the coefficients which control the linear and the exponential isotropic hardening laws and, σ_∞ is the temperature dependent saturation flow stress parameter.

The visco-plastic strains, $\boldsymbol{\varepsilon}^{vp}$ are considered purely deviatoric $\boldsymbol{\varepsilon}^{vp} = \mathbf{e}^{vp}$. Their definition can be derived from the principle of maximum plastic dissipation together with the evolution law of the isotropic strain-hardening variable, ξ, as it is written in EQUATIONS 3.42 and 3.43.

$$\dot{\mathbf{e}}^{vp} = \dot{\gamma}^{vp} \frac{\partial \Phi(\mathbf{s}, q, T)}{\partial \mathbf{s}} = \dot{\gamma}^{vp} \mathbf{n} \tag{3.42}$$

$$\dot{\xi} = \dot{\gamma}^{vp} \frac{\partial \Phi(\mathbf{s}, q, T)}{\partial q} = \dot{\gamma}^{vp} \sqrt{\frac{2}{3}} \tag{3.43}$$

where $\mathbf{n} = \frac{\partial \Phi(\mathbf{s},q,T)}{\partial \mathbf{s}} = \frac{\mathbf{s}}{\|\mathbf{s}\|}$ is the normal to the yield surface. The visco-plastic multiplier, $\dot{\gamma}^{vp}$, is computed assuming a rate-dependent evolution law as it is shown in EQUATION 3.44.

$$\dot{\gamma}^{vp} = \left\langle \frac{\Phi(\mathbf{s}, q, T)}{\eta} \right\rangle^{\frac{1}{m}} \tag{3.44}$$

where $\langle \cdot \rangle$ represents the Macaulay brackets. Therefore, it is possible to experience a viscous overstress, $\eta(\dot{\gamma}^{vp})^m$, is allowed to exceed the yield surface, being $\eta(T)$ the temperature dependent plastic viscosity, and $m(T)$ its temperature dependent rate sensitivity. The resulting equivalent stress yields:

$$\sigma_{eq} = \sqrt{\frac{3}{2}} [f_S(T) R + \eta(\dot{\gamma}^{vp})^m] \tag{3.45}$$

Depending on the temperature the material is in a certain phase: solid, mushy or liquid-like. For each of these phases, there are some particularities in the formulation which are to be explained separately.

Solid phase

In the case of solid phase, $T < T_S$ and $f_S(T) = 1$, is straight-forward: (eqs. 3.46 and 3.47)

$$p = K(e^{vol} - e^T) \tag{3.46}$$

$$\mathbf{s} = G(\mathbf{e} - \mathbf{e}^{vp}) \tag{3.47}$$

where the evolution laws for the visco-plastic strains and isotropic hardening are deduced from EQUATIONS 3.42 and 3.43 as for the classical *J2-thermo-elasto-visco-plastic* model.

Liquid-like phase

The liquid phase is characterize by a temperature field above the liquidus temperature, $T > T_L$ and $f_S = 0$. In the liquid-like phase, the elastic and

thermal strains do not develop. Therefore, it is considered an isochoric (incompressible) behaviour: $e^{vol} = 0$ which reduces to the continuity equation for mass conservation that reads as follows in EQUATION 3.48.

$$\nabla \cdot \mathbf{u} = 0 \tag{3.48}$$

On the other hand, all the deformations are purely deviatoric (viscous): $\mathbf{e} = \mathbf{e}^{vp}$. In fact, the yield surface radius defined in EQUATION 3.40 reduces as the temperature increases till vanishing above the liquidus temperature. Inserting this result into the evolution law of the plastic multiplier in EQUATION 3.44, it is possible to recover a purely viscous model of the following form (3.49):

$$\mathbf{s} = \eta \dot{\mathbf{e}}^{vp} \tag{3.49}$$

in other words, a *Newtonian* law depending on the viscous strain-rate.

Mushy phase

THe material model should provide a smooth transition between the solid and the liquid like phase. The ranges of solid fraction and temperatures are: $0 < f_S < 1$ and $T_S < T < T_L$, respectively. The material behaviour of the mushy phase is described by a *non-Newtonian* flow ($m \neq 1$) of the form shown in EQUATION 3.50.

$$\mathbf{s} = \eta(\dot{\boldsymbol{\varepsilon}}^{vp})^m \tag{3.50}$$

3.2.4 Finite element modeling

The finite element modeling is not a trivial feature of additive manufacturing. In contrast to other kind of simulations, such as the mechanical analysis of bridges, dams, or buildings; the target domain is not constant in time, that is, the printed part is growing in each time step and, therefore, its boundary is also evolving in time. Additionally, the source term of the energy balance equation ($\dot{Q}$) [10] is moving along a user-defined scanning sequence.

Activation algorithm

In order to differentiate between elements regarding the role they are playing in each time step, an activation strategy called *born-dead-elements technique* is employed [1, 18]. This strategy consists of classifying the elements in three types: active, inactive and activated. Before starting the printing process, all the elements belonging to the AM material part are inactive. In each time step, according to the scanning sequence and the desired heat affected zone (HAZ), a certain number of elements will be activated using a searching algorithm. These elements represent the advancing metal deposition layer, as well as, the melting pool. Once an element which has been activated abandons the HAZ, their type changes to active. Only active and activated elements are computed and assembled into the global matrix. In figure 3.3, the role of each element during the simulation of additive manufacturing of a box is presented.

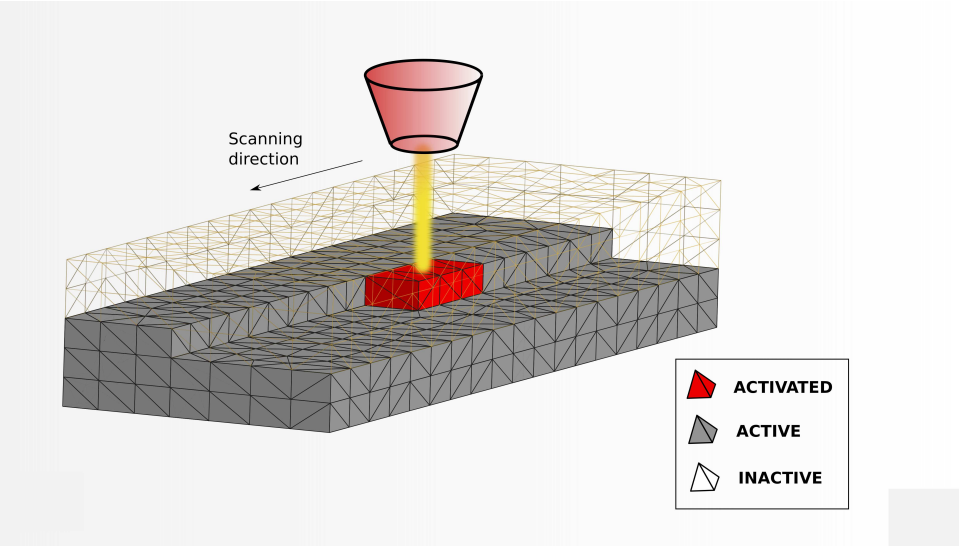


FIGURE 3.3: Activation algorithm

As it can be observed, the elements which have been already affected by the laser (grey elements) are active, while the elements which have not played a role yet are inactive (transparent elements). In this particular time step, the laser is affecting a certain area which elements are activated (red elements)[2].

Space-time discretization

Actually, the laser beam in additive manufacturing follows a continuous path. However, in the discrete problem with time step $\Delta t = t^{n+1} - t^n$, the melting pool moves along the scanning path stepping from time t^n to time t^{n+1}. As a result, the power input is intermittent, it can be alleviate by reducing the time step, but not avoided. Regarding the space discretization, even if the heat affected zone (HAZ) is discretized by means of a very fine mesh, it is very challenging to achieve enough spatial resolution to define exactly the ideal volume where the power input must be introduced. Therefore, thanks to the searching algorithm, it is possible to sum the volume of all the elements belonging to the melting pool which reads as:

$$V^{pool} = \sum_{e=1}^{n_e} V^{(e)} \Big|_{e \in pool} \tag{3.51}$$

Once the melting pool volume is computed, the (average) density distribution of the heat source rate (per unit of volume) can be written as:

$$\dot{Q} = \frac{\eta_p \dot{P}}{V^{pool}} \tag{3.52}$$

where $\dot{P}$ is the total energy rate input introduced by the laser and η is the heat absorption efficiency. This power redistribution preserves the total energy input regardless the mesh employed. Additionally, special care must be focus on computing the heat dissipation though the boundaries, always taking into account that the domain is growing, the boundary is

[2] In figure 3.3, the penetration is not taken into account

changing every single time-step and, therefore, an updating algorithm is to be performed in order to capture these evolutions.

3.2.5 Definition of the scanning path

As it was mentioned in the previous section, the laser follows an user-defined scanning path. This scanning path is defined by means of a universal format called Common Layer Interface (CLI). CLI is employed as an input of geometry data to model fabrication systems based on layer manufacturing technologies [18]. Its simplicity and flexibility allows its use for a wide range of applications such as layer-wise photo-curing of resins, sintering of powder material, solidification, medical scanner, and any other systems which build models on a layer-by-layer manner.

The 3D geometry is sliced with parallel planes in the *z-axis* direction generating the so-called layers. Each layer is defined by its thickness and a set of contours and hatches. Contours denote the boundaries of the solid geometry within the layer and are defined by polylines. On the other hand, a hatch is a set of independent straigh lines, each one defined by a start and an end point, which purpose is defining filling structures to obtain the solid model. For the sake of understandability, a simple example is provided in FIGURE 3.4.

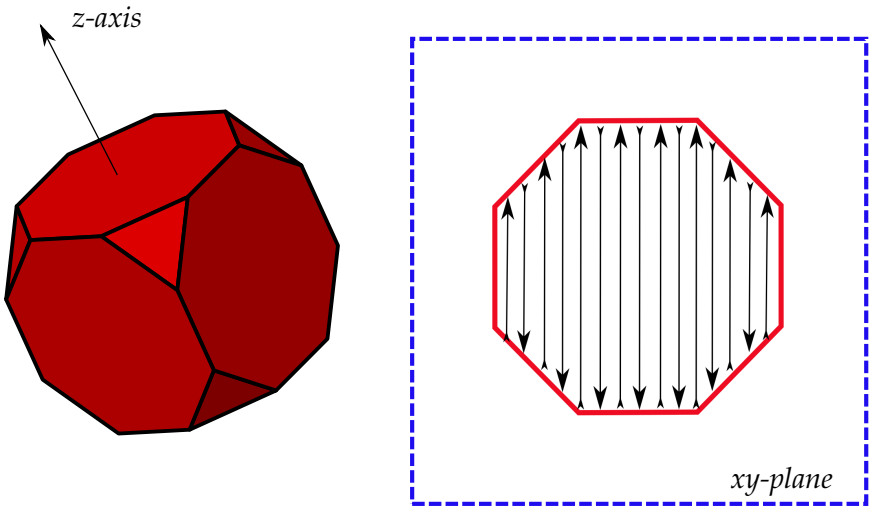


FIGURE 3.4: 3D model slicing example

A truncated cube is to be sliced and, for instance, one of its octagonal sections is chosen. The discontinuos (blue) line represent the *xy-plane* employed to slice the geometry, the continuous (red) line represents the contour (polyline) of the solid material (it could be curve line), and the solid black arrows are the hatches, with their start and end point, used to fill the layer. In this case the filling of the layer is full, as it is done in the numerical simulations of this dissertation, however, numberless possibilities exist when choosing the scanning sequence.

In the CLI file, there is no reference to the time scale. The movement of the laser-head along the hatches is specified in the general data file by three parameters: the scanning speed (v_{MD}) when the laser is printing the layer,

the back-speed (v_{back}) when the laser is switch-off and repositioning, and a fix advancing step (Δs) which is the distance gone across a certain hatch in one time step. Using these three parameters, the program computes automatically the time steps employed in the printing and repositioning processes, Δt_{on} and Δt_{off} respectively. These time steps are calculated in the following way:

$$\Delta t_{on} = \frac{\Delta s}{v_{MD}} \tag{3.53}$$

$$\Delta t_{off} = \frac{\Delta x}{v_{back}} \tag{3.54}$$

where Δx is computed taking into account the end point of the hatch n and the starting point of the hatch $n + 1$. It is crucial to take into account Δt_{off} as part of the thermal analysis because of the fast cooling experienced by the material in this time.

Chapter 4

Methodology

In this chapter, the complete procedure to perform the experimental calibration and sensitivity analysis is explained. In other words, starting from the importance of achieving these analyses, the work-flow followed to reach the results which includes: election of parameters and finite-elements model, work strategy, auxiliary tasks, among others, is presented in detail.

4.1 Experimental calibration

The experimental calibration of the additive manufacturing (AM) thermo-mechanical problem is performed comparing the numerical results with an experimental campaign carried out at the *State Key Laboratory of Solidification Processing* (SKLSP). The thermal part of the process is already calibrated [18], therefore, the material parameters for the thermal part, as well as, the welding ones are correctly calibrated and are taken as starting point of the mechanical calibration. First of all, a brief explanation about the experimental campaign and the numerical model which is chosen to compare with the experimental results is performed.

4.1.1 Experimental campaign

The experimental calibration took place at the *State Key Laboratory of Solidification Processing* (SKLSP) employing the laser solid forming (LSF-III) machine (FIGURE 4.1) [18]. The system uses a CO_2-laser source with maximum power input of 4 kW in a close chamber with a protected atmosphere (argon) to prevent oxidation. The comparison with the experimental results is performed comparing temperature data and vertical displacements. The temperature is measured using OMEGA GG-K-30 thermocouples and a midi LOGGER G900-4/8 data recorder.

The substrate samples consists of a *TI6Al4V* Titanium alloy plates, 140 mm long, 50 mm wide and 6 mm thick. Each plate is burnished by sand paper and cleaned using anhydrous-alcohol and acetone. Later on, five thermocouples are spot-welded: three on the lower surface and two on the upper surface. For the data comparison, the lower surface CH1 thermocouple is employed, as well as, the vertical displacement (VD) gauge located over the lower surface and with coordinates: $x = 0.055y = 0.02z = -0.006$ (m) (FIGURE 4.2). Finally, the base is clamped to a suitable supporting system inside the AM chamber.

The scanning sequence to print each layer consists of a set of horizontal hatches changing the direction in each one with respect the previous one as it is shown in FIGURE 4.3. Repeating this sequence ten times, it is possible

FIGURE 4.1: LSF-III machine

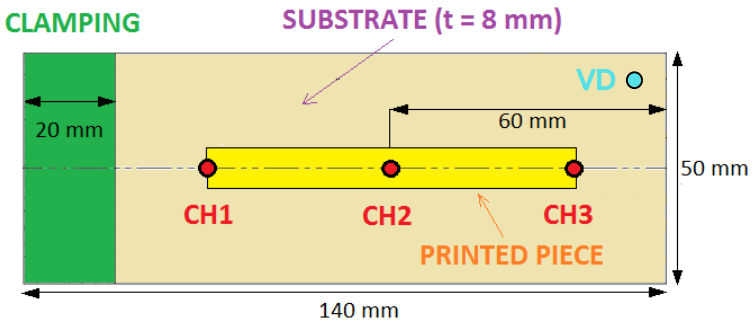


FIGURE 4.2: Experiment thermocouple location and vertical displacement gauge

to print each of the layers of added material. In TABLE 4.1 the process parameters used to inform the LSF machine are collected.

PARAMETER	VALUE	UNIT
Power input	2	kW
Laser beam size	1.6	mm
Offset distance	0.8	mm
Up-lift height	0.3	mm
Scanning speed	10	mm/s
Back speed	50	mm/s
Powder feeding	8.5	r/min

TABLE 4.1: Process parameters used in the LSF machine

4.1.2 Computational model

The computational model for the correct reproduction of the experimental campaign and, therefore, the appropriate experimental calibration of the computational framework, should take into account the following aspects: geometry, boundary conditions, material characterization, and laser features. In the following sections, a brief explanation of each part including some clarifying figures is presented.

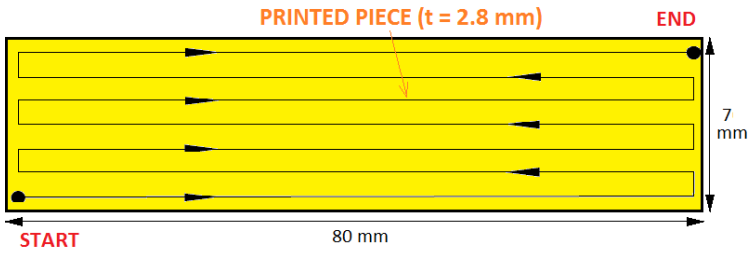


FIGURE 4.3: Scanning sequence of the AM part

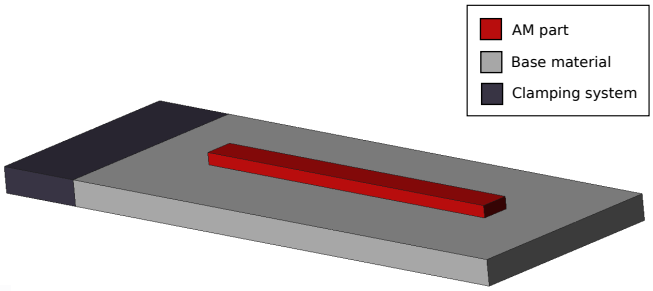


FIGURE 4.4: Computational model for numerical simulation

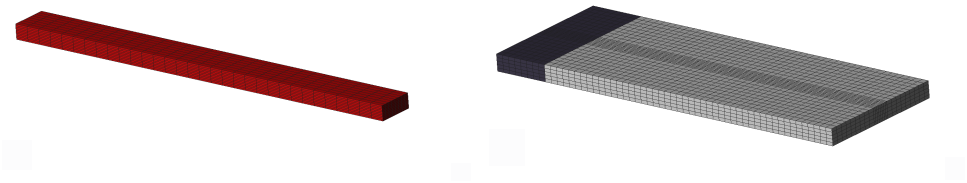
Geometry and Mesh

The geometry is simple to reproduce due to the fact that it is basically three parallelepiped pieces located one after the others. In FIGURE 4.4, the three material parts of the simulation are presented: the base and the clamping system which supports the process, and the additive manufacturing part which is to be printed. The dimensions of the different parts are: clamping system 20x50x6 mm, base material 120x50x6 mm, and the AM part 80x7x2.8 mm.

The finite element mesh employed to compute the simulation is composed by linear hexahedra. The mesh size chosen for the AM part is not a trivial parameter. It should provide enough resolution, at least in the *z-direction*, to capture each deposited layer. In the *xy-plane*, the size is not so crucial, however, a finer mesh would return better result. The mesh size in the AM part is strictly linked to the welding parameters. If high-fidelity (element by element) simulation is desired, the activation technique would work in an element by element manner, in other words, a finer mesh would increase the computational cost in each time step and, indirectly, a greater number of time steps would be introduced. Therefore, increasing the resolution of the mesh increases the problem size both in space and time.

As it can be observed in FIGURE 4.5, the mesh which is in command is the AM part. Once this mesh is designed, the basement and clamping meshes are less restrictive and a greater mesh size can be chosen. The mesh

size is chosen in order to respect each one of the ten deposited layer and keeping a reasonable value of the accuracy and the computational time.



(A) AM part (B) Basement and clamping

FIGURE 4.5: Finite element mesh

Boundary conditions

The boundary conditions imposed in the simulation are purely mechanical. The upper and lower surfaces of the clamping system are constrained in the *z-direction*. Additionally, the clamping system exterior corners have their movements fully constrained in order to ensure the well-posedness of the problem. The thermal problem is controlled by the initial temperature of activation of the AM part and the energy introduced in the system by the heat source, however, explicit thermal constraints are not defined.

Material characterization

The same material is employed for both the basement and the powder to print the additive manufacturing part. The material is a Titanium-alluminium alloy (Ti6Al4) in which the temperature dependence of the mechanical and thermal parameters is taken into account [13]. Therefore, it is necessary to define the set of material parameters in a wide temperature range and, additionally, the heat loss properties of the material including the main heat dissipation mechanism presented in the state of the art (3.2.1). As it was mentioned several times in the previous sections, the starting point of the experimental calibration was the correct calibration of the thermal part of the problem [18]. Therefore, the parameters related to the thermal phenomenon such as: thermal conductivity, density, specific heat, the heat transfer coefficients by conduction, convection... are properly calibrated and they are taken as known in the mechanical experimental calibration of this dissertation. The main challenge is to play with the parameters related to the mechanical part in order to capture correctly the vertical displacement data collected in the experimental campaign. These parameters are mainly: Young's modulus, Poisson's ratio, thermal dilatancy, and plastic properties of the material. The material properties which are fixed during this experimental calibration are collected in the APPENDIX A.

Welding parameters

The welding parameters employed to compute the additive manufacturing are the ones extracted from the LSF machine specified in 4.1.1. However, for the definition of the scanning sequence and the energy absorption of the system, an appropriate assumption should be performed. In the case of the scanning sequence, the distance advanced by the laser in each time step should be determined and, due to the fact that high-fidelity additive manufacturing is desired, this distance coincides with the size of the mesh in the movement direction of the laser. With respect to the uncertain energy absorption coefficient, the value inherited from the thermal calibration is adopted. The welding parameters, as well as, the definition of the scanning sequence in each layer are included in the APPENDIX B.

4.1.3 Simplified model

During the first attempts to reproduce the mechanical part (vertical displacements) of the additive manufacturing experimental calibration, it was observed that the displacements belonging to the printing stage were recovered following the tendency of the experimental data. However, the cooling phase was inaccurate, with different shape in the numerical and the experimental curve. Due to this fact, a simplified model was created in order to rapidly run simulations varying the parameters trying to find the source of error. The finite element model selected to perform this analysis must fulfill certain features. First of all, it should be computationally cheap. A huge number of models are to be run combining the different parameters with different values and, therefore, the model should run as quick as possible. Secondly, the model should be suitable in order to appreciate the effect of the parameters variation and resembles as much as possible to the real problem employed for the experimental calibration. In order to reach these requirements, a quasi-2D model composed of linear hexahedral elements is chosen. This model tries to simplify the full high-fidelity simulation which is to be experimentally calibrated by reducing the thickness to the whole part to an element. In FIGURE 4.6, the simplified model is presented.

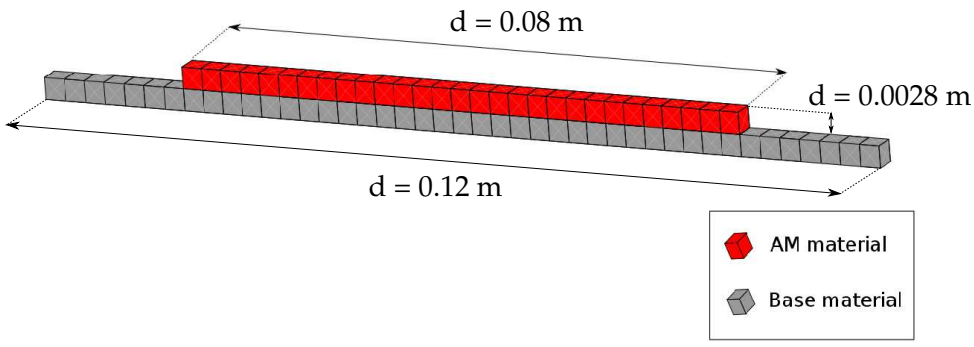


FIGURE 4.6: Sensitivity analysis finite-elements model

The model is clamped on the left surface in the base material. Both additive manufacturing (AM) and base material are the same: an alloy of

titanium and aluminum. The full specification of the material, as well as, their temperature dependent properties are collected in the appendix A.

4.1.4 Python subroutines

The results which are to be extracted are the vertical displacement of the free tip of the base material and the temperature evolution of the middle point over the deposited layer. The procedure to run and plot a huge number of cases is automatically performed thanks to the implementation of three python subroutines [5] which basically computes the following tasks:

- Preparation of the suitable data files to be run.
- Computation of the simulations.
- Plotting of the different results to compare and extract a conclusion.

This subroutines are included in APPENDIX C.

Subroutine 1: Data file creator

For each value of the parameter or combination of parameters involved in the sensitivity analysis, a data file should be created. For instance, if the material density influence is studied, it would be necessary to create as data files as values of the density are desired. Creating this huge number of data files one by one and saving them with a suitable, coherent name is a tiring and madding task. Therefore, to avoid this fruitless task, data file creator subroutine was created. To use it, three simple steps must be followed:

1. Create the desired model and save it with a desired name, for instance: *modelo.dat*
2. In *modelo.dat*, change the value of the target parameter for: *VALUE1*
3. Open the python script and introduce the name of the desired parameter and the list of values assigned to this parameter.

The script will generate as models as values of the parameter are in the list. Each of these models receive a name following the pattern: 'modelo*NAMEVALUE*.dat', for instance in the case of a value 1500 of the density, the name would be: *modeloDENSI1500.dat*. It is clear the convenience of employing this subroutine both in terms of saving time and keeping an order in the nomenclature of the files. A similar script is implemented in order to change two parameters of the model. This script combine the whole parameters of the two list defined.

Subroutine 2: Executable list

Once the collection of data files is created, it is time to run the simulations. This subroutine simply creates a *cmd* executable file, collecting the models generated with subroutine 1 and preparing them to be run automatically one after the other. The main advantages of this subroutine are:

- Avoiding the necessity to run the models one by one.
- Let the computer works without human-supervision.
- Avoid the oblivion of running any case.

Subroutine 3: Selective plot generator

The last step consists of plotting the results. The selective plot generator creates a *.txt* file directly used by the open source program *gnuplot* [27]. In this file, it includes the plots specified in the plot name. In order to select the files to plot, regular expressions are employed [9]. It must be taken into account the fact that sometimes the simulation generates more than one plot (for example, temperature and displacements), and the user can decide whether plot one or the other. For instance, if the temperature plot is created with the extension *.cur01* and the displacement with *.cur02*, to generate the *.txt* plot file drawing the temperature of all the models, the following regular expression must be written: '.*.cur01'.

4.1.5 Visco-elastic material response

The experimental calibration is based on modifying the material parameters in order to match the numerical results with the experimental campaign ones. However, it was observed during the first simulations, as it is commented in section 4.1.3, that the cooling part of the experimental curve was not recovered. The problem encountered was that the shape of the curve was not reproducible, therefore an extra ingredient was necessary in order to provide the material model with the capability to capture the real problem. This ingredient was the introduction to the material model of a visco-elastic response. Therefore, the material model specified in the state-of-the-art which was *J2-thermo-elasto-visco-plastic* contitutive model will be updated to *J2-thermo-visco-elasto-visco-plastic* constitutive one [11].

The visco-elastic material response is taken into account introducing a rheological model based on springs and dashpots. The model employed will be the so-called Standard Solid Model which is composed by a spring and Kelvin-Voigt solid connected in a series.

Standard solid model

The standard solid model[1] is a three-parameter model (E_1, E_2, and η) and it is used to represent the behaviour of a number of materials (FIGURE 4.7).

The models introduces two elastic components (E_1 and E_2), and a viscosity one (η). First, the elastic element (E_1) absorbs the initial deformation, however, as the Kelvin-Voigt starts to work, the energy gradually accumulates in the second spring (E_2) controlling this process the progressive energy dissipation induced by the dashpot. It can be defined the so-called retardation time which can reads as follows in EQUATION 4.1.

$$\tau = \frac{\eta}{E_2} \tag{4.1}$$

[1] http://courses.washington.edu/bioen520/notes/Viscoelasticity.pdf

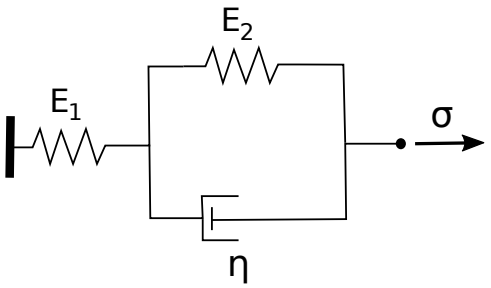


FIGURE 4.7: Standard Solid Model

As it can deduced, the model has an instantaneous elastic response thanks to the spring E_1, followed by a visco-elastic response controlled by the retardation time leading to a system which behaves as two springs in series.

The Standard solid model constitutive equation can be written in terms of the three material parameters (E_1, E_2, and η), the total stress (σ), the total strain (ε), and its rates ($\dot{\sigma}$ and $\dot{\varepsilon}$, respectively). Considering that ε_1, ε_2, and ε_3 are the strains of the two spring elements and the dashpot respectively, and, σ_1, σ_2, and σ_3 their respective stresses. Since element 1 is connected in series to elements 2 and 3 which are in parallel, the following equations can be deduced.

$$\sigma = \sigma_1 = \sigma_2 + \sigma_3 \tag{4.2}$$
$$\varepsilon = \varepsilon_1 + \varepsilon_2 \tag{4.3}$$
$$\varepsilon_2 = \varepsilon_3 \tag{4.4}$$

The elemental constitutive equations reads as follows:

$$\sigma_1 = E_1\varepsilon_1 \tag{4.5}$$
$$\sigma_2 = E_2\varepsilon_2 \tag{4.6}$$
$$\sigma_3 = \eta\dot{\varepsilon_3} \tag{4.7}$$

The stress resulting in the Kelvin-Voigt model can be written as follows in EQUATION 4.8.

$$\sigma = E_2\varepsilon_2 + \eta\dot{\varepsilon_3} \tag{4.8}$$

Plugging EQUATIONS 4.6 and 4.8 into EQUATION 4.4, the total strain can be written as follows in EQUATION 4.9.

$$\varepsilon = \frac{\sigma}{E_1} + \frac{\sigma}{E_2 + \eta(\frac{d}{dt})} \tag{4.9}$$

Employing cross multiplication and rearranging terms, the final constitutive model is written in EQUATION 4.10.

$$(E_1 + E_2)\sigma + \eta\dot{\sigma} = E_1E_2\varepsilon + E_1\eta\dot{\varepsilon} \tag{4.10}$$

For infinite time or nule viscosity, the equation of two springs in parallel is recovered. The so-called elastic modulus for infinite time (E_∞) can be written in the following way:

$$E_\infty = \frac{E_1 E_2}{E_1 + E_2} = \beta E_1 \tag{4.11}$$

The infinite ratio β ($0 < \beta < 1$) is employed in the code to reduce the value of the stiffness from E_1 to E_∞ controlled by the retardation time τ. Due to the fact that, in the code, the volumetric and deviatoric parts of the stress tensor are computed separately, this visco-elastic constitutive model can be directly implemented to these parts of the stress tensor. Consequently, the model is provided with a initial bulk and shear moduli (K_0 and G_0) which absorbs the initial deformation, and a retardation time (τ_K and τ_G) which controls the visco-elastic behaviour until reaching the infinite value of the system (K_∞ and G_∞) which can be written as follows in EQUATIONS 4.12 and 4.13.

$$K_\infty = \frac{K_0 K_2}{K_0 + K_2} = \beta_K K_0 \tag{4.12}$$

$$G_\infty = \frac{G_0 G_2}{G_0 + G_2} = \beta_G G_0 \tag{4.13}$$

The values of the infinite ratios for the bulk and shear moduli affect directly the value of the Young modulus and the Poisson ratio. Whereas the Young modulus has not a set of admissible values, Poisson ratio is restricted to the interval $0 < \nu < 0.5$. Therefore, the set of values which can be imposed to the infinite ratios, β_K and β_G, is not completely free. The expressions of the Young modulus and the Poisson ratio in terms of the bulk and shear moduli affected by their respective infinite ratios read as follows in EQUATIONS 4.14 and 4.15.

$$E = \frac{9\beta_K K \beta_G G}{3\beta_K K + \beta_G G} \tag{4.14}$$

$$\nu = \frac{3\beta_K K - 2\beta_G G}{2(3\beta_K K + \beta_G G)} \tag{4.15}$$

Imposing the inequality for the Poisson ratio in the expression 4.15, the following restrictions for the infinite ratios, β_K and β_G, are obtained.

$$\frac{\beta_K}{\beta_G} > \frac{2}{3}\frac{G}{K} \Longrightarrow \nu > 0 \tag{4.16}$$

$$\beta_G > 0 \Longrightarrow \nu < 0.5 \tag{4.17}$$

Inequality 4.17 is trivial due to the hypothesis in the value of the infinite ratio. However, inequality 4.16 states a limit for the ratio between the infinite ratios in which the Poisson ratio will remain positive and, consequently, a real material model is simulated. It is possible to plot the variation of the Young modulus and the Poisson ratio when varying the infinite ratio parameters (FIGURES 4.8 and 4.9).

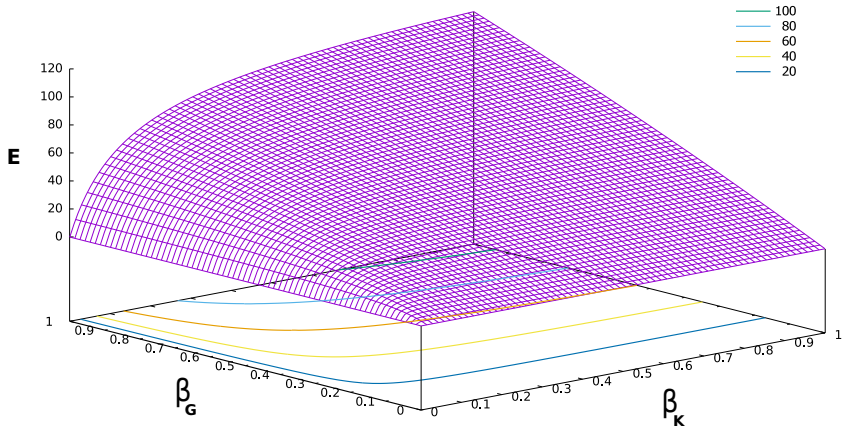


FIGURE 4.8: Young modulus as function of the infinite ratios

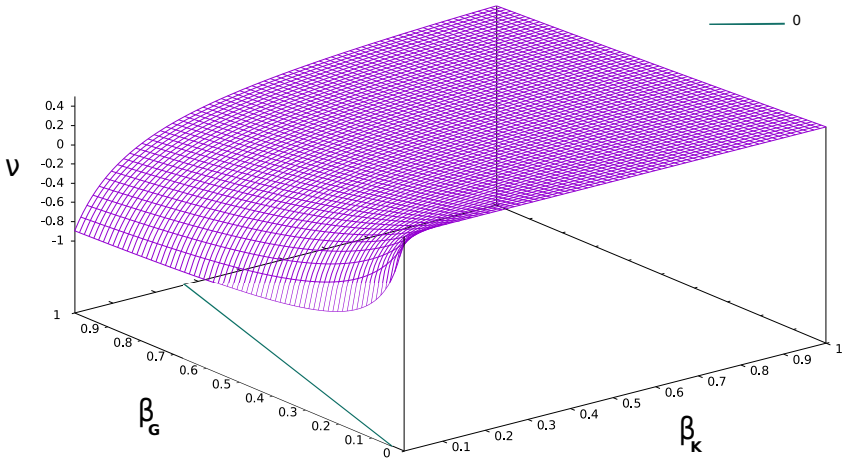


FIGURE 4.9: Poisson ratio as function of the infinite ratios

In FIGURE 4.9, it can be observed the blue line contour in base which indicates the combinations of the infinite ratio values for which the Poisson ratio is equal to zero. Therefore, the blue line determines the limit below which the material behaves non-real ($\nu < 0$).

4.2 Sensitivity analysis

The sensitivity analysis of certain features of the computational framework for the additive manufacturing simulation by blown powder is performed in order to assess the mechanisms which play an important role in the calculations. The computational model employed in this sensitivity analysis is the same used in the experimental calibration, therefore, the model development included in 4.1.2 can be inherited in this part. Moreover, the python code written for taking optimal advantage of the simplified model

is recycled in this section in order to rapidly generate the models to be run (4.1.4, 4.1.4, and 4.1.4).

The features which are to be analyzed in this sensitivity analysis are:

- Importance of plasticity for the correct matching of the experimental results.
- Importance of the visco-elasticity implemented in 4.1.5.
- Importance of moving the temperature range of the so-called mushy phase.

Starting from the well-calibrated model, the result employed to appreciate the importance of each analysis is the vertical displacement captured in the experimental campaign in the VD gauge (FIGURE 4.2). The parameters which are modified are purely mechanical, therefore, the result in terms of temperatures should remain identical for any election of the sensitivity values.

4.2.1 Sensitivity to plasticity

The plasticity is known to be crucial in the correct calibration of the model. In the experimental calibration, all the materials (support and AM part) are considered to behave plastically when the stresses in the material reach the yield limit and, consequently, the material experiences permanent deformations beyond this value. The plasticity sensitivity analysis is divided into two steps. First of all, the different combinations of elastic and plastic material assumptions are considered. These combinations are collected in TABLE 4.2, leading to four different situations: one purely elastic, one purely plastic and two mixed scenarios.

	Elastic support	**Plastic support**
Elastic AM part	Pure elastic	Mixed
Plastic AM part	Mixed	Pure plastic

TABLE 4.2: Plasticity combinations

Secondly, the hardening effect is evaluated. The sensitivity to varying the initial yield stress in which the plastification starts, the different isotropic hardening parameters (linear and exponential), and modifying the final maximum value of the yield surface is studied.

4.2.2 Sensitivity to visco-elasticity

The visco-elasticity is known to be a mean mechanism to reproduce the cooling phase as it is shown in the simplified model during the experimental calibration (4.1.3). However, a deeper analysis in the real computational model is performed in order to capture this effect in the additive manufacturing phase and achieving a more representative result for the comparison. The parameters which can be tackled in this part of the sensitivity analysis are those presented in 4.1.5 which are basically the infinite ratios (β_K and β_G).

4.2.3 Sensitivity to the phase change temperature range

The material is considered to behave as a solid and, furthermore, solid theory is employed in this dissertation. However, as it is mentioned in the state of the art 3.2.3 [17], for a value of the temperature over a so-called liquidus temperature, T_{Liq}, no volumetric strains are considered and the material is supposed to behave purely deviatoric. This phase change range, between a solid temperature and a liquidus temperature, is subjected to be modified in order to assess the influence of the phase change in the calculations [26]. In the particular case of the titanium, the phase change can lead to a material phase transformation which can vary the internal structure of the material and, consequently, its properties [8]. The initial temperature for the melting pool, in which the laser energy is focused, is set to the solidus temperature due to the fact that it is considered the material melting when it is injected. Therefore, in order to achieve the sensitivity to the phase change hypothesis, different cases are run varying the solidus and liquidus temperatures, T_{Sol} and T_{Liq} respectively, but keeping constant the mushy phase range of temperatures, $T_{Liq} - T_{Sol}$.

Chapter 5

Results and Discussion

In this chapter, the results and the discussion about the sensitivity analysis and the experimental calibration are shown. First of all, the experimental calibration is presented comparing the numerical results with the experimental campaign carried out at SKLSP laboratories. Once the model is properly calibrated, the sensitivity analysis, specified in the methodology, is performed.

5.1 Experimental calibration

The experimental calibration is an iterative process varying the material parameters which are supposed to influence the results in the model. In this case, the first step was studying the starting point of the task. As it was commented in the methodology, the thermal part of the problem was successfully calibrated. Therefore, the next step is to evaluate the mechanical part. The results provided by the SKLSP laboratories are the vertical displacement of the titanium basement in a certain point defined in the methodology. First of all, a first simulation attempt without varying any parameter of the correct thermal calibration in order to know the starting point of the mechanical one is performed. Secondly, taking as origin this previous simulation, the *apropos* hypothesis and assumptions should be done in order to match the numerical simulation with the experimental campaign data.

5.1.1 Starting point: Successfully calibrated thermal problem

The thermal part were correctly calibrated as it is commented in the methodology. Therefore, it is possible to assure the correct thermal distribution for the computation of the mechanical part which is the aim of this experimental calibration. In FIGURE 5.1, the temperature comparison between the numerical simulation and the experimental data for the three thermocouples described in 4.1.1 is shown. Additionally, in FIGURE 5.2 a detailed view of the printing phase is provided in order to appreciate the accurate reproduction of the thermal oscillations.

From this point forward, the results which are to be shown are those related to the mechanical part. In this case, these results are the vertical displacement in the VD gauge specified in 4.1.1. In the classical *isothermal split* [15], the coupled system of equations is partitioned into a thermal phase at fixed configuration, followed by a mechanical phase at constant temperature. The thermal result (temperature) should not differ between the cases taking into account that the thermal parameters are not modified

and there is no influence of the mechanical part into the thermal part, that is, the problem is only coupled in the thermal to mechanical direction.

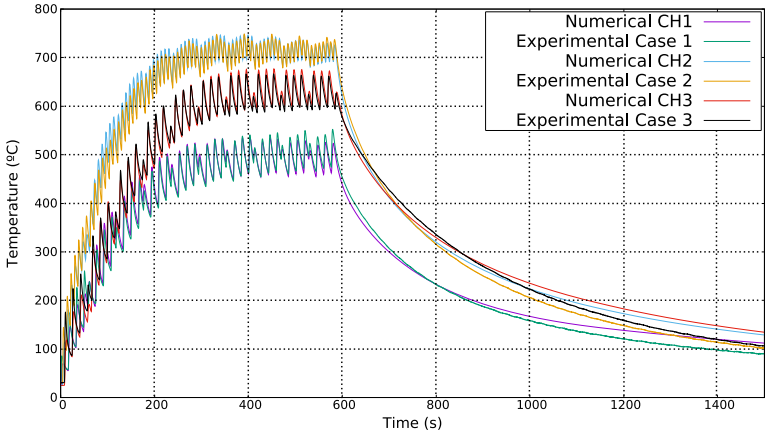


FIGURE 5.1: Accurate thermal results

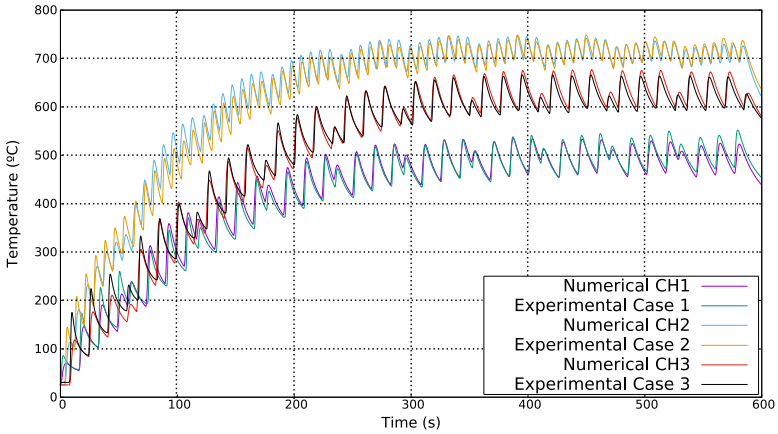


FIGURE 5.2: Accurate thermal results (printing detail)

5.1.2 Mechanical calibration: initial model

The starting point of the mechanical calibration is the correct thermal one. The unmodified model returns a vertical displacement plot that matches the experimental data in the additive manufacturing stage but differs noticeably in the cooling phase as it can be observes in FIGURE 5.3.

There are two main differences between the experimental data and the numerical simulation which are:

- Rapid deformation in the beginning of the cooling phase, not showing the mild slope till reaching the final deformation.
- Appearance of a peak which seems to be unnatural in a monotonous material cooling.

On the other hand, the final vertical displacement is correctly captured, as well as the starting point of the cooling phase. These facts give the idea of a correct calibration of the global deformation of the problem. A series of analysis were performed in the simplified computational model presented

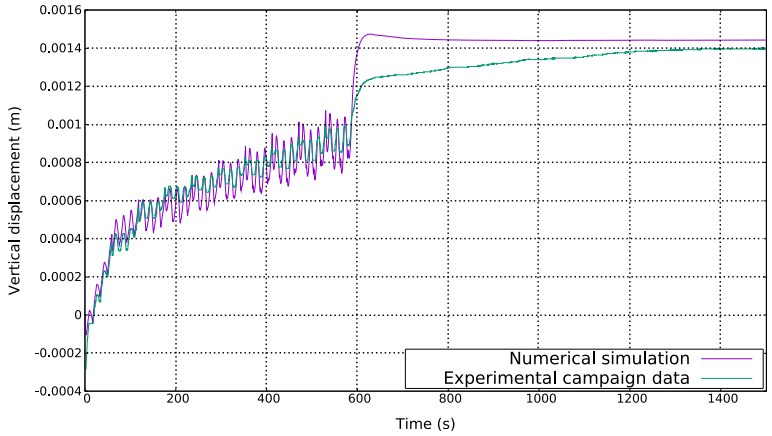


FIGURE 5.3: Starting point for the mechanical calibration

in 4.1.3 in order to assess the material parameters which can be tackled affecting the results. The material parameters which are supposed of playing a role in the simulation are: Young's modulus (E), Poisson's ratio (ν), plasticity parameters, and thermal dilatancy (α).

5.1.3 Mechanical calibration: Poisson's ratio

The first parameter which is to be calibrated is the Poisson's ratio (ν). It is observed in the simplified model that this parameter has a big influence over the result. In FIGURE 5.4, the temperature dependent Poisson's ratio in the original model and in the calibrated one is presented.

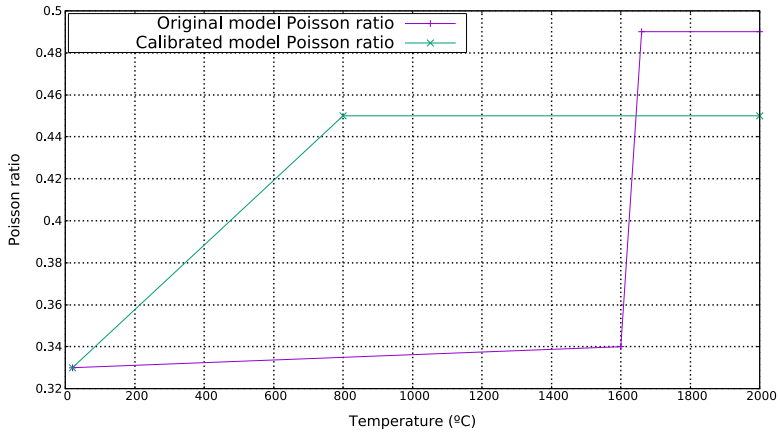


FIGURE 5.4: Original and calibrated Poisson's ratio

The original calibration assumed an almost constant value of the Poisson's ratio in the range of temperatures under the melting point (1660 ^0C) followed by a sharp increment to 0.49 when exceeding this melting temperature trying to simulate the liquid-like phase. However, as it is observed later on, there is a critic temperature (800 ^0C), far beneath the melting point, over which the material behaves liquid-like and, consequently, the Poisson's ratio is imposed to be near 0.5 (incompressibility condition) which characterizes the fluid behaviour. In FIGURE 5.5 can be observed the effect of performing this variation of the Poisson's ratio. Calibrating the Poisson's

ratio on base of the assumptions explained previously, it is possible to avoid the appearance of the peak in the vertical displacement.

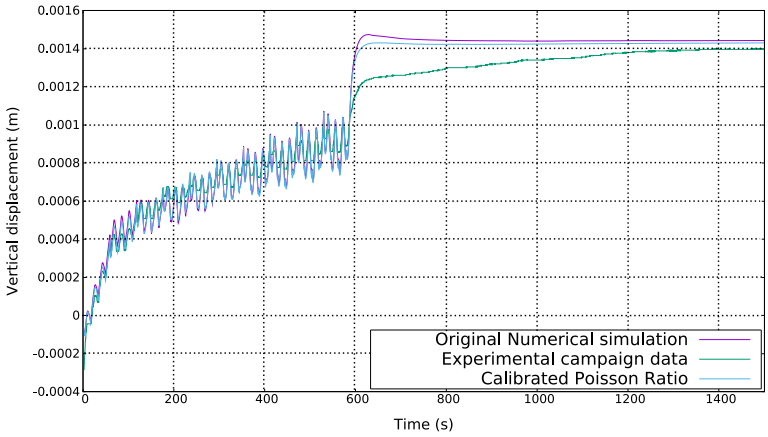


FIGURE 5.5: Effect of Poisson's ratio calibration

Modifying parameters which have a linear effect on the deformation such as the Young's modulus or the thermal dilatancy is not the way to reproduce the experimental data due to the fact that the result would be a translation of the obtained result with the calibrated Poisson's ratio. Therefore, the insertion of a time-dependent model which takes into account the retardation in the vertical displacement is necessary.

5.1.4 Mechanical calibration: Visco-elastic behaviour

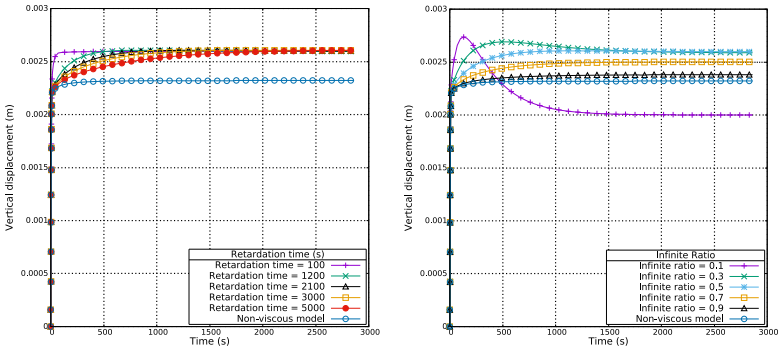
The Standard solid model developed in 4.1.5 induces a visco-elastic behaviour for the volumetric part (bulk modulus, K) and for the deviatoric part (shear modulus, G). The parameters which must be defined in the model are the so-called infinite ratio (β_K and β_G) and the retardation time (τ_K and τ_G). Briefly, the idea is that for a time equals to the retardation time, the value of the bulk and shear moduli were close to their so-called infinite values which read as: $K_\infty = \beta_K K_0$ and $G_\infty = \beta_G G_0$ being K_0 and G_0 the temperature dependent value of the bulk and shear moduli for the intact material.

In order to know the effect of introducing this visco-elastic behaviour in the material model, the simplified example is employed to run a large number of simulations.

Visco-elastic effect in simplified model

The effect of visco-elasticity for both varying the visco-elastic parameters for the volumetric part and for the deviatoric one is presented. Knowing the effect provoked to the model by these variables, it would be simpler recovering the experimental campaign data. First of all, the effect of the visco-elasticity over the bulk modulus is observed in FIGURE 5.6.

Constant values of the infinite ratio (β_K) lead to the same final vertical displacement. Additionally, as it can be observed in FIGURE 5.6a, the retardation time slows down the convergence to this final value. In FIGURE 5.6b, different values of the infinite ratio for the same retardation time are



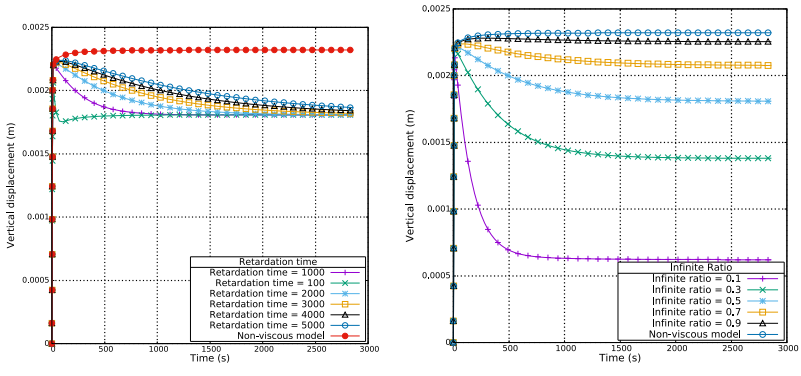
(A) $\beta_K = 0.5$ (B) $\tau_K = 1800s$

FIGURE 5.6: Sensitivity to bulk modulus visco-elasticity

tried. The solution increases as the infinite ratio decreases due to the fact the material is losing stiffness, however, for a very low value of this infinite ratio, the curve is under the intact model, the reason is that the Poisson's ratio is reduced until it reaches negative values as it is explained in 4.1.5. Therefore, the material is behaving as a non-real metal or metamaterial. The limit for a positive Poisson's ratio as it can be deduced from EQUATION 4.16 is:

$$\beta_K > \frac{2}{3}\frac{G}{K} \tag{5.1}$$

Secondly, analogous analysis is to be performed with the visco-elasticity of the shear modulus. In this case, there is no restriction for the infinite ratio of shear modulus (β_G) to keep positive the Poisson's ratio and, therefore, all the solutions obtained in the sensitivity analysis of the shear visco-elasticity represents feasible natural materials. The same plots as in the bulk modulus case are presented. First varying the retardation time keeping constant the infinite ratio (β_G) and, secondly, modifying the infinite ratio and keeping constant the retardation time.



(A) (B)

FIGURE 5.7: Sensitivity to shear modulus visco-elasticity

In FIGURE 5.7a, for a constant value of the infinite ratio $\beta_G = 0.5$ the retardation time (τ_G) determines the rate of convergence to the stationary solution which coincides with $G = G_\infty$. On the other hand, in FIGURE 5.7b, for a constant value of the retardation time, it can be observed that the

final vertical displacement decreases when decreasing the infinite ratio for the shear modulus (β_G).

The information extracted from this sensitivity analysis for the visco-elastic scheme in the simplified model can be directly employed in the mechicanical experimental calibration of the real problem.

Visco-elastic effect in real problem

The next step consists of matching the experimental data in the cooling phase. As it can be observed in FIGURE 5.5, it is necessary to decrease the vertical displacements at the beginning of the cooling and increase them gradually until the end of the simulation. Thanks to the results extracted in 5.1.4, it is known the effect produced by introducing the visco-elastic response in the material constitutive model. For the final correct calibration of the numerical simulation, it is necessary to state several assumptions for the visco-elastic behaviour. First of all, in the liquid-like material phase, the Poisson's ratio, ν, is near to its limit value 0.5. In this case, as $\nu \longrightarrow 0.5$, the bulk modulus, K, tends to infinity (incompressibility constraint) and, a purely viscous scheme is considered (3.2.3). Therefore, in this viscous scheme, the parameter which plays a role is the shear modulus, however, its value must be also softened during the printing phase. In order to achieve this goal, reasonable values of the infinite ratio, β_G, and its retardation time, τ_G, are defined. Secondly, once the material is cooling, the volumetric part (controlled by the bulk modulus) starts to work again. There is a certain softening after the printing process in the value of the bulk modulus and, therefore, its infinite ratio, β_K, should be calibrated. The correct calibrated model has the values included in TABLE 5.1.

	Calibrated model	**Non visco-elastic model**
Infinite ratio β_K	0.7	1
Infinite ratio β_G	0.4	1
Retardation time τ_K	6000	0
Retardation time τ_G	600	0

TABLE 5.1: Calibrated visco-elastic parameters

In TABLE 5.1, the hypothesis in the bulk and shear moduli visco-elasticity are reflected in the parameter values. Consequently, a faster and higher softening is experience by the shear modulus, while the bulk modulus loses its stiffness during the cooling phase. It must be remarked that, during the printing, the vanishing of the volumetric deformation contribution avoid the need for paying attention to the visco-elastic values of the bulk modulus. The final result of the experimental calibration, comparing the starting point, the calibrated model and the experimental data, is presented in FIGURE 5.8.

As it can be observed in FIGURE 5.8, the printing stage is not very influenced by the visco-elasticity, returning a result similar to the starting point simulation. However, during the cooling phase, the final vertical displacement result is reproduced accurately and, furthermore, the initial behaviour of the curve, avoiding the peak or the excessive elastic recovering, and the

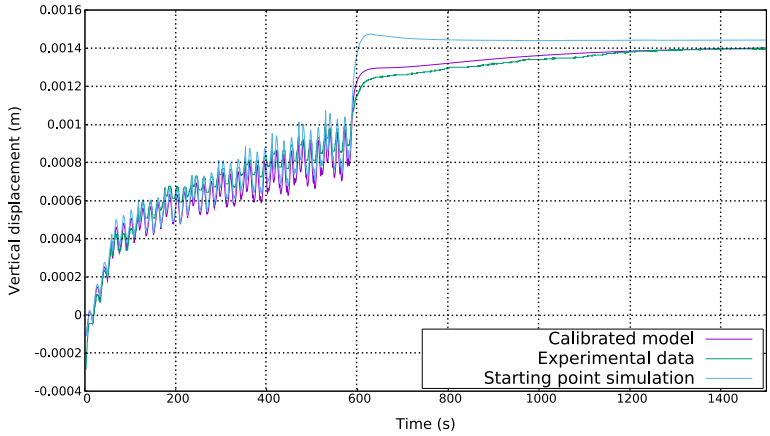


FIGURE 5.8: Calibrated model

mild slope until reaching the total vertical displacement is successfully reproduced. It would be possible to find more accurate values of the viscoelastic parameters in order to fit closely the experimental data. However, each simulation lasts five hours and the presented final result is supposed to satisfy the aim of the experimental calibration in terms of accuracy.

5.2 Sensitivity Analysis

The sensitivity analysis is performed in order to obtain a better understanding of the phenomena taking place in the additive manufacturing process. Taking into account that the model is well-calibrated for the vertical displacements, this result is the one which is to be compared between each simulation varying the features collected in the methodology (4.2).

5.2.1 Sensitivity to plasticity

The sensitivity to plasticity is performed in two steps as it is presented in the methodology. First, the four combinations collected in TABLE 4.2 are run in order to assess the differences between pure elastic and pure plastic, and a mixed of both in the support and the AM part. Secondly, once the importance of plasticity is shown, a sensitivity analysis varying the hardening of the scheme is done.

Elasticity and plasticity comparison

The verticalmente displacement results for the four different combinations of the constitutive model of the AM part and the support collected in TABLE 4.2 is presented in FIGURE 5.9.

Thanks to this comparison, it is possible to appreciate the effect of plasticity to the vertical displacements in the simulation. Regarding the printing stage, there are two clear tendencies. On the one hand, those with plastic AM part present a contribution to the vertical displacement appreciable during the printing phase leading to a linear tendency of the average value of the oscillations. On the other hand, those with elastic AM part, remain horizontal tendency during the oscillations and do not add any permanent

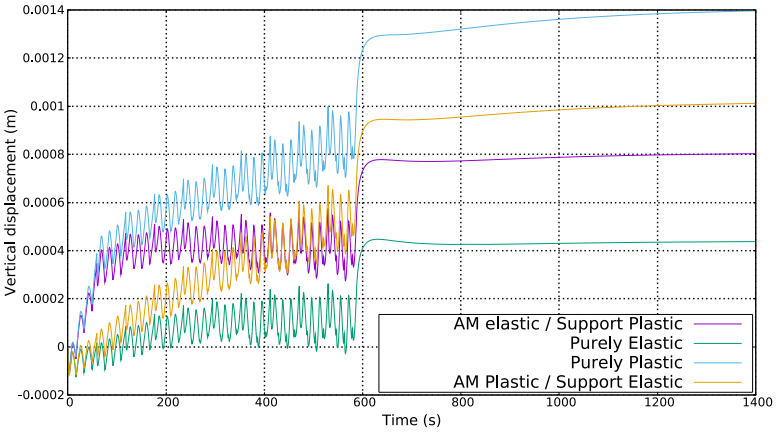


FIGURE 5.9: Elasticity and plasticity comparison

vertical displacement to the total one. The plasticity contribution of the support is observable during the initial stage of the printing phase. There is a high permanent deformation during the deposition of the first layer which induces sharp vertical displacements. The result is not surprising taking into account the high temperature gradient provoked during the first layer when the base material, at room temperature, receives the high energy laser input.

Sensitivity to hardening

Once the importance of taking into account the plasticity phenomenon is shown, the sensitivity to modifying the main plastic parameters involved in the simulation is required. As it is mentioned in the methodology (4.2.1), these parameters are:

- The initial and final values of the yield stress which represents the yield surface.
- The linear isotropic hardening parameter.

First of all, different values for the initial yield stress and its correspondent saturation flow stress are tried. In order to set the values of these parameters, two cases are to be run increasing and decreasing the aforesaid parameters up to 20% with respect to the values of the experimentally calibrated model. These two cases are collected in TABLE 5.2.

	YIELD STRESS (20/600 ºC)	**SAT. FLOW STRESS (20/600 ºC)**
CASE 1	$763 \cdot 10^6$ / $396 \cdot 10^6$	$1144 \cdot 10^6$ / $595 \cdot 10^6$
CASE 2	$923 \cdot 10^6$ / $516 \cdot 10^6$	$1384 \cdot 10^6$ / $775 \cdot 10^6$

TABLE 5.2: Initial yield and saturation flow stress cases ($\frac{N}{m^2}$)

In FIGURE 5.10, the sensitivity to the initial yield stress and the saturation flow stress is shown. Although plasticity constitutive model assumption is crucial for the correct matching of the experimental data with the numerical results, the slightly significant variation of the initial yield stress does not return remarkable different results. As it can be seen, the solutions for the printing phase are closed each others, being a little bit more

observable at the cooling phase due to the accumulation of plastic strains which leads to a higher vertical displacement. The tendency is clear, as the yield stress is reduced, the material plastifies before and, therefore, higher vertical displacements are experienced.

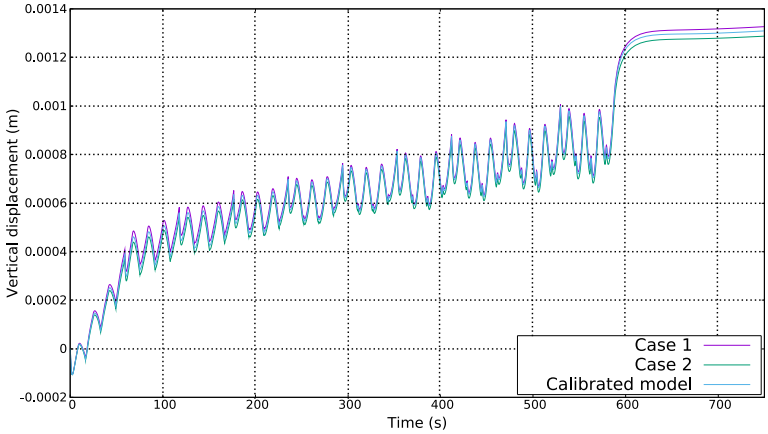


FIGURE 5.10: Sensitivity to initial yield and saturation flow stresses

Secondly, txhe importance of varying the linear isotropic hardening parameter, which is the responsible of allowing the material to increase the radius of its yield surface and, therefore, experience higher plastic strains, is assessed by running several simulations with the temperature dependent values of the coefficient collected in TABLE 5.3.

TEMPERATURE (ºC)	CASE 1	CASE 2	CASE 3
20	800 e+04	800 e+08	800 e+08
600	300 e+04	300 e+04	300 e+08

TABLE 5.3: Linear hardening combination cases

The results of these three cases are shown in FIGURE 5.11. The main difference is observed when varying the linear hardening coefficient for high temperature (600 ^oC). On the one hand, the lower is this parameter for high temperature (Cases 1 and 2), the more plastic strains are generated and, therefore, higher vertical displacements. Additionally, it can be appreciate that the average slope of the printing stage is steeper as the linear hardening coefficient decreases. On the other hand, the value of the linear hardening parameter for low temperatures is not important. It can be observed, comparing cases 1 and 2, that the only difference is a little variation of the vertical displacement in the cooling phase due to the low temperature plastification of the material. As additional remark, it can be commented the higher plastification during the first layer which, as it was shown in 5.2.1, is provoked by the support plastification.

5.2.2 Sensitivity to visco-elasticity

The numerical simulations during the experimental calibration, both in the simplified model and in the real model, proved that the visco-elasticity material behaviour is an important aspect to achieve an accurate reproduction of the experimental data. Therefore, a deeper sensitivity analysis is to be

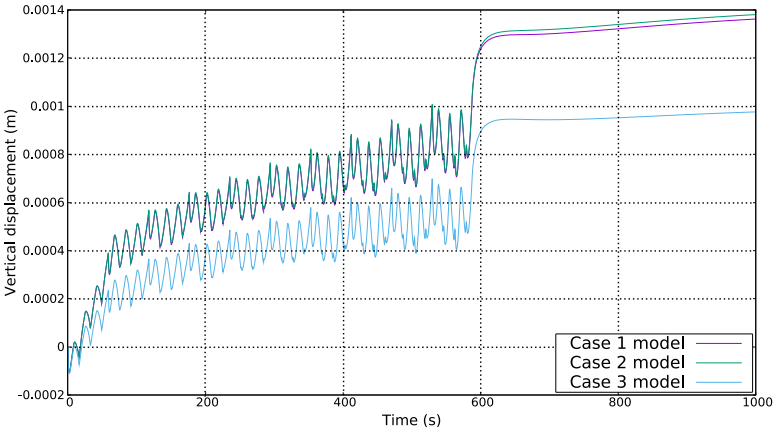


FIGURE 5.11: Sensitivity to linear isotropic hardening coefficient

carried out to determine the influence of each visco-elastic parameter in the simulation results. In this case, two different cases are taking into account. The variation of both the infinite ratios for the bulk and shear moduli keeping constant the retardation time.

Sensitivity to bulk modulus infinite ratio β_K

In FIGURE 5.12, the vertical displacement results for different simulations of the real problem varying the infinite ratio for the bulk modulus is performed. As it was advanced in the sensitivity analysis carried out with the simplified model, reducing the value of β_K leads to greater vertical displacements due to the material softening. At the beginning of the process, the effect of visco-elasticity is still unnoticeable and the four cases collide, however, while the time evolves, this effect is gradually perceptible reaching its highest difference during the cooling phase.

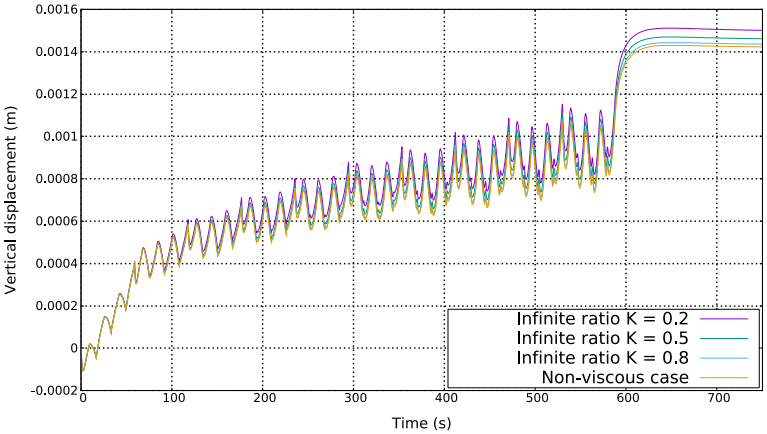


FIGURE 5.12: Sensitivity to infinite ratio β_K

Sensitivity to shear modulus infinite ratio β_G

The sensitivity to shear modulus visco-elasticity works analogously to the bulk modulus one but with contrary effect in the vertical displacement.

Consequently, a decrease in the infinite ratio β_G leads to lower vertical displacements as it can be observed in FIGURE 5.13. The effect of viscoelasticity is negligible in the early steps, reaching gradually its highest perception at the cooling phase. It can be pointed out that, for the same value of the infinite ratio, the shear modulus has a major influence in the final result than the bulk modulus. The reasoning to explain this higher influence of shear modulus resides in the fact that the liquid-like phase is characterized by a purely deviatoric (viscous) behaviour and, therefore, for certain ranges of the temperature, only the shear modulus is playing a role in the simulation.

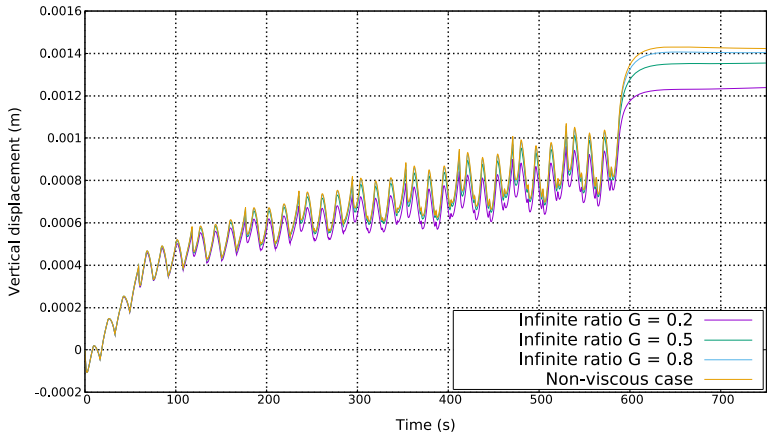


FIGURE 5.13: Sensitivity to infinite ratio β_G

5.2.3 Sensitivity to phase change temperature range

The sensitivity analysis to phase change temperature is performed by mean of four situations. The different scenarios, which are collected in TABLE 5.4, have different values for the liquidus and solidus temperatures keeping constant the mushy phase range ($T_{liq} - T_{sol}$).

	SOLIDUS T^a	LIQUIDUS T^a
CASE 1	600	660
CASE 3	1000	1060
CASE 4	1200	1260

TABLE 5.4: Phase change cases

The results in FIGURE 5.14 show the importance of knowing the range of temperatures where the phase change occurs. As it can be observed, the higher is the phase change range of temperatures, the greater vertical displacement is achieved at the end of the simulation (printing + cooling). The reason of this result is the possibility of the material to deform in a higher range of temperatures, taking into account that the volumetric strain is neglected above the value of the liquidus temperature, T_{liq}. On the other hand, at the beginning of the simulation (during the first layer), the case with lower phase change range experiences the steepest vertical displacement. As it was commented in 5.2.1, the responsible of the vertical displacements in the first stage of the simulation is the support and, concretely, the

plastification of it. The explanation to this phenomenon is that the additive manufacturing part, subjected to great temperatures, rapidly behaves liquid-like and, therefore, does not contribute to the final result in terms of plastification. In other words, the whole plastification is experienced by the support leading the higher vertical displacements at the beginning for the lower solidus and liquidus temperatures.

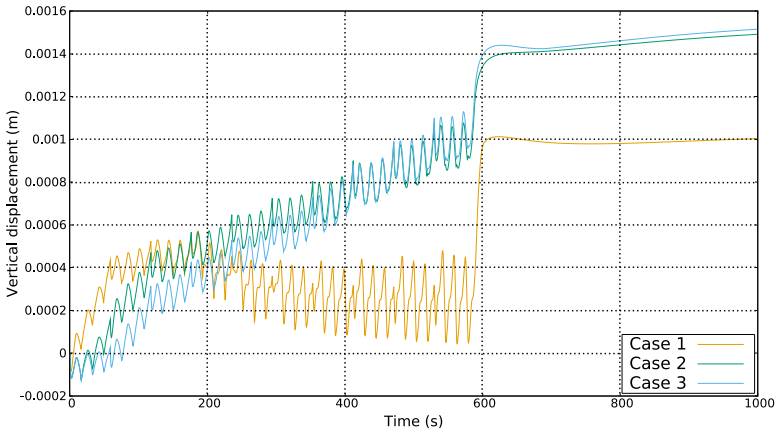


FIGURE 5.14: Sensitivity to phase change range

Chapter 6

Conclusions

In this dissertation, a finite element computational framework for the numerical simulation of additive manufacturing (AM) by blown powder is calibrated through experimental data. Additionally, a sensitivity analysis is performed. Theoretical aspects such as: heat transfer and stress analysis, constitutive model, finite element activation algorithm, among others, are presented. Furthermore, a visco-elastic scheme, which has been developed and implemented in order to successfully calibrate the model, is also detailed.

The experimental calibration of the mechanical results has been carried out comparing the numerical simulations with the experimental data provided by the SKLSP laboratories in China. These simulations have shown a remarkable agreement with the experimental evidence during the printing stage from the beginning of the work. However, the cooling phase required two major changes to be suitably calibrated. Firstly, the material is considered to be incompressible when reaching the critical temperature (800 ^oC). Incompressibility is characterized by a Poisson's ratio equal to 0.5 ($\nu = 0.45$ in the simulation). Secondly, a visco-elastic scheme is implemented in order to capture the softening phenomenon after the deposition of the material. This scheme is able to decrease the instantaneous vertical displacements (elastic recovering) at the beginning of the cooling and distribute them along all the stage. With these two enhancements, the experimental data is correctly reproduced.

The sensitivity analysis is focused on three important aspects of the simulation: the material plasticity, the visco-elastic behaviour, and the temperature range of the phase change. The outcome of this analysis is the following:

- Plasticity is the main source of permanent deformations. Both the support and AM part plastifications are crucial for performing a correct and accurate simulation. The support plastification leads to a sudden high vertical displacements at the beginning of the printing whereas the AM part plastification distributes the vertical displacements during all the scanning process giving a certain slope to the oscillating printing stage.

- The simulation is not very sensitive to variations (up to 20%) in the plasticity parameters. However, increasing the hardening of the material, both in the support and the AM part, produces less permanent deformation and, therefore, lower vertical displacements.

- Visco-elasticity is essential to capture the cooling phase. The softening of the shear modulus decreases the mechanical behaviour of the vertical displacement whereas the softening of the bulk modulus increases it. Combining these two mechanism with suitable values of the retardation time permits the reproduction of AM cooling taking into account the material damping effect.

- The phase change range of temperatures establishes the limit above which the material behaves purely deviatoric, without volumetric deformations. The higher this change range is considered, the larger vertical displacements are experienced due to the plastification of the AM part. However, the steep vertical displacements at the beginning of the process are gradually lost due to the minor contribution of the support to the plastic deformation.

Lines for future work

Finally, this point outlines the possible future developments in the thermo-mechanical analysis of additive manufacturing by blown powder arising as a natural continuation of the present work. The following research lines are suggested:

- The constitutive model do not take into account the phase change crystallization which is known to cause an important variation in the material properties. The visco-elastic behaviour deals with this phenomenon during the cooling phase but it is not enough. An implementation of an algorithm which differentiates the elements depending on their crystal structure and assigns them the suitable material properties is pending.

- The material properties are considered to be macroscopic although the crystal structure is mentioned to be an important issue. Therefore, an interesting idea of research is the definition of thermo-physical material properties (at macro scale) which depend on the metallurgical evolution (at micro scale).

- The computational framework is run activating element-by-element (high-fidelity). This is computationally expensive and, in most real cases, it is unfeasible for industrial applications. An interesting research line is the study of a layer-by-layer or hatch-by-hatch activation. It is shown that these techniques return an accurate average value of the temperatures (no oscillations). Knowing this thermal fact and trying to expect somehow the plastification induced by the oscillations, it could be possible to capture accurately the mechanical behaviour.

Bibliography

[1] Lundbäck A. "Finite element modeling and simulation of welding of aerospace components". In: *Department of Applied Physics and Mechanical Engineering, Luleå University of Technology, Lule å* (2003).

[2] Thomas Campbell et al. "Could 3D printing change the world". In: *Technologies, Potential, and Implications of Additive Manufacturing, Atlantic Council, Washington, DC* (2011).

[3] Celentano D., Oller S., and Oñate E. "A coupled thermomechanical model for the solidification of cast metals". In: *International journal of solids and structures* 33.5 (1996), pp. 647–673.

[4] Bugeda G. et al. "Numerical analysis of stereolithography processes using the finite element method". In: *Rapid Prototyping Journal* 1.2 (1995), pp. 13–23.

[5] Van Rossum G. and Drake Jr F. *Python reference manual*. Centrum voor Wiskunde en Informatica Amsterdam, 1995.

[6] Campbell I., Bourell D., and Gibson I. "Additive manufacturing: rapid prototyping comes of age". In: *Rapid Prototyping Journal* 18.4 (2012), pp. 255–258.

[7] Gibson I., Rosen D., Stucker B., et al. *Additive manufacturing technologies*. Springer, 2010, pp. 19–41.

[8] Elmer J. et al. "Phase transformation dynamics during welding of Ti–6Al–4V". In: *Journal of applied physics* 95.12 (2004), pp. 8327–8339.

[9] Friedl J. *Mastering regular expressions*. " O'Reilly Media, Inc.", 2002.

[10] Goldak J., Chakravarti A., and Bibby M. "A new finite element model for welding heat sources". In: *Metallurgical transactions B* 15.2 (1984), pp. 299–305.

[11] Simo J. and Hughes T. *Computational inelasticity*. Vol. 7. Springer Science & Business Media, 2006.

[12] Lamport L. *Latex*. Addison-Wesley, 1994.

[13] Lindgren L. et al. "Simulation of additive manufacturing using coupled constitutive and microstructure models". In: *Additive Manufacturing* (2016).

[14] Cervera M., Agelet de Saracibar C., and Chiumenti M. "COMET: COupled MEchanical and Thermal analysis. Data input manual". In: *Barcelona: International Center for Numerical Method in Engineering (CIMNE)* (2002).

[15] Cervera M., Agelet de Saracibar C., and Chiumenti M. "Thermo-Mechanical analysis of industrial solidification processes". In: *Rapid Prototyping Journal* ().

[16] Cervera M. et al. "Mixed linear/linear simplicial elements for incompressible elasticity and plasticity". In: *Computer Methods in Applied Mechanics and Engineering* 192.49 (2003), pp. 5249–5263.

[17] Chiumenti M. et al. “Numerical modeling of the electron beam welding and its experimental validation”. In: *Rapid Prototyping Journal* ().

[18] Chiumenti M. et al. “Numerical simulation and experimental calibration of Additive Manufacturing by blown powder technology. Part I: thermal analysis”. In: *Rapid Prototyping Journal* ().

[19] Abedrabbo N., Pourboghrat F., and Carsley J. “Forming of AA5182-O and AA5754-O at elevated temperatures using coupled thermo-mechanical finite element models”. In: *International Journal of Plasticity* 23.5 (2007), pp. 841–875.

[20] Higham N. *Handbook of writing for the mathematical sciences*. Siam, 1998.

[21] Zienkiewicz O. and Taylor R. *The finite element method*. Vol. 1-4. McGraw-hill London, 1977.

[22] Mudge R. and Wald N. “Laser engineered net shaping advances additive manufacturing and repair”. In: *WELDING JOURNAL-NEW YORK-* 86.1 (2007), p. 44.

[23] Ribó R. et al. *Gid, the personal pre and post processor*. Tech. rep. Technical report, 1997.

[24] Warren M Rohsenow, James P Hartnett, Young I Cho, et al. *Handbook of heat transfer*. Vol. 3. McGraw-Hill New York, 1998.

[25] Huang S. et al. “Additive manufacturing and its societal impact: a literature review”. In: *The International Journal of Advanced Manufacturing Technology* 67.5-8 (2013), pp. 1191–1203.

[26] Agelet de Saracibar C., Cervera M., and Chiumenti M. “On the formulation of coupled thermoplastic problems with phase-change”. In: *International journal of plasticity* 15.1 (1999), pp. 1–34.

[27] Williams T. and Kelly C. *gnuplot—an interactive plotting program (official gnuplot manual)*. 2004.

[28] Wong K. V and Hernandez A. “A review of additive manufacturing”. In: *ISRN Mechanical Engineering* 2012 (2012).

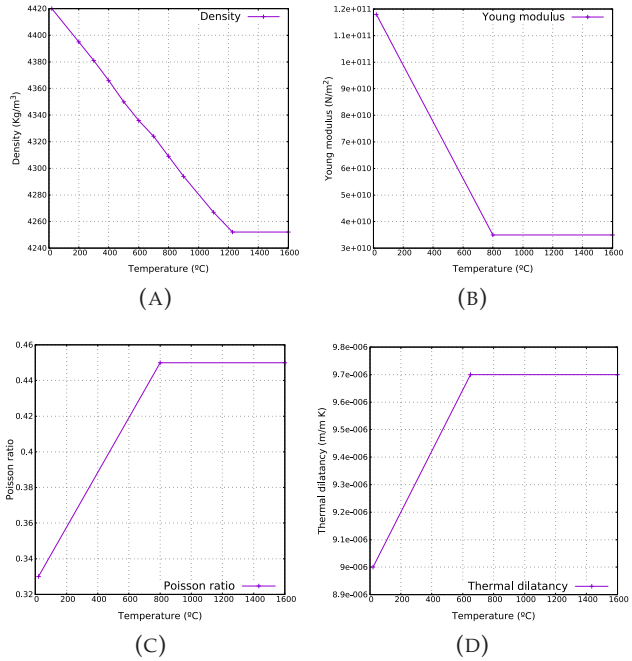
[29] Huang W. and Lin X. “Research progress in laser solid forming of high-performance metallic components at the state key laboratory of solidification processing of china”. In: *3D Printing and Additive Manufacturing* 1.3 (2014), pp. 156–165.

[30] Zhai Y., Lados D., and LaGoy J. “Additive manufacturing: making imagination the major limitation”. In: *JOM* 66.5 (2014), pp. 808–816.

Appendix A

Material specification

A.1 Mechanical parameters

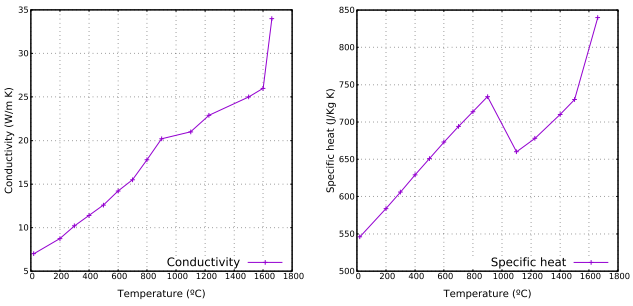


(A) (B)

(C) (D)

FIGURE A.1: Mechanical parameters

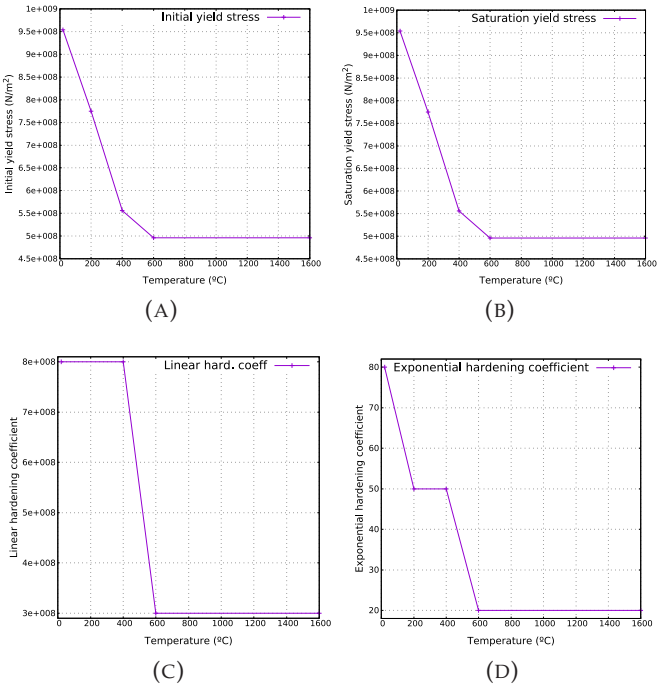
A.2 Thermal parameters



(A) (B)

FIGURE A.2: Thermal parameters

A.3 Plasticity parameters



(A) (B)

(C) (D)

FIGURE A.3: Plasticity parameters

Appendix B

Welding specification

B.1 Welding parameters

PARAMETER	VALUE	UNIT
Power input	2	kW
Absorption Coefficient	0.138	
Advancing step	2.5	mm
Layer thickness	0.28	mm
Scanning speed	9.8	mm/s
Back speed	50	mm/s
Laser base	1.75	mm
Laser penetration	2	mm

TABLE B.1: Welding parameters

B.2 Scanning sequence

The scanning sequence is defined by a CLI format file which reads as follows:

```
$$HEADERSTART
$$ASCII
$$UNITS/1
$$VERSION/200
$$LAYERS/1
$$HEADEREND
$$GEOMETRYSTART
$$LAYER/0.0
$$HATCHES/1 1  -40.0 -2.625  40.0 -2.625
$$HATCHES/1 1   40.0 -1.750 -40.0 -1.750
$$HATCHES/1 1  -40.0 -0.875  40.0 -0.875
$$HATCHES/1 1   40.0  0.000 -40.0  0.000
$$HATCHES/1 1  -40.0  0.875  40.0  0.875
$$HATCHES/1 1   40.0  1.750 -40.0  1.750
$$HATCHES/1 1  -40.0  2.625  40.0  2.625
$$LAYER/0.28
$$HATCHES/1 1  -40.0 -2.625  40.0 -2.625
$$HATCHES/1 1   40.0 -1.750 -40.0 -1.750
$$HATCHES/1 1  -40.0 -0.875  40.0 -0.875
$$HATCHES/1 1   40.0  0.000 -40.0  0.000
```

```
$$HATCHES/1 1  -40.0  0.875  40.0  0.875
$$HATCHES/1 1   40.0  1.750 -40.0  1.750
$$HATCHES/1 1  -40.0  2.625  40.0  2.625
$$LAYER/0.56
$$HATCHES/1 1  -40.0 -2.625  40.0 -2.625
$$HATCHES/1 1   40.0 -1.750 -40.0 -1.750
$$HATCHES/1 1  -40.0 -0.875  40.0 -0.875
$$HATCHES/1 1   40.0  0.000 -40.0  0.000
$$HATCHES/1 1  -40.0  0.875  40.0  0.875
$$HATCHES/1 1   40.0  1.750 -40.0  1.750
$$HATCHES/1 1  -40.0  2.625  40.0  2.625
$$LAYER/0.84
$$HATCHES/1 1  -40.0 -2.625  40.0 -2.625
$$HATCHES/1 1   40.0 -1.750 -40.0 -1.750
$$HATCHES/1 1  -40.0 -0.875  40.0 -0.875
$$HATCHES/1 1   40.0  0.000 -40.0  0.000
$$HATCHES/1 1  -40.0  0.875  40.0  0.875
$$HATCHES/1 1   40.0  1.750 -40.0  1.750
$$HATCHES/1 1  -40.0  2.625  40.0  2.625
$$LAYER/1.12
$$HATCHES/1 1  -40.0 -2.625  40.0 -2.625
$$HATCHES/1 1   40.0 -1.750 -40.0 -1.750
$$HATCHES/1 1  -40.0 -0.875  40.0 -0.875
$$HATCHES/1 1   40.0  0.000 -40.0  0.000
$$HATCHES/1 1  -40.0  0.875  40.0  0.875
$$HATCHES/1 1   40.0  1.750 -40.0  1.750
$$HATCHES/1 1  -40.0  2.625  40.0  2.625
$$LAYER/1.40
$$HATCHES/1 1  -40.0 -2.625  40.0 -2.625
$$HATCHES/1 1   40.0 -1.750 -40.0 -1.750
$$HATCHES/1 1  -40.0 -0.875  40.0 -0.875
$$HATCHES/1 1   40.0  0.000 -40.0  0.000
$$HATCHES/1 1  -40.0  0.875  40.0  0.875
$$HATCHES/1 1   40.0  1.750 -40.0  1.750
$$HATCHES/1 1  -40.0  2.625  40.0  2.625
$$LAYER/1.68
$$HATCHES/1 1  -40.0 -2.625  40.0 -2.625
$$HATCHES/1 1   40.0 -1.750 -40.0 -1.750
$$HATCHES/1 1  -40.0 -0.875  40.0 -0.875
$$HATCHES/1 1   40.0  0.000 -40.0  0.000
$$HATCHES/1 1  -40.0  0.875  40.0  0.875
$$HATCHES/1 1   40.0  1.750 -40.0  1.750
$$HATCHES/1 1  -40.0  2.625  40.0  2.625
$$LAYER/1.96
$$HATCHES/1 1  -40.0 -2.625  40.0 -2.625
$$HATCHES/1 1   40.0 -1.750 -40.0 -1.750
$$HATCHES/1 1  -40.0 -0.875  40.0 -0.875
$$HATCHES/1 1   40.0  0.000 -40.0  0.000
$$HATCHES/1 1  -40.0  0.875  40.0  0.875
$$HATCHES/1 1   40.0  1.750 -40.0  1.750
$$HATCHES/1 1  -40.0  2.625  40.0  2.625
```

```
$$LAYER/2.24
$$HATCHES/1 1  -40.0 -2.625  40.0 -2.625
$$HATCHES/1 1   40.0 -1.750 -40.0 -1.750
$$HATCHES/1 1  -40.0 -0.875  40.0 -0.875
$$HATCHES/1 1   40.0  0.000 -40.0  0.000
$$HATCHES/1 1  -40.0  0.875  40.0  0.875
$$HATCHES/1 1   40.0  1.750 -40.0  1.750
$$HATCHES/1 1  -40.0  2.625  40.0  2.625
$$LAYER/2.52
$$HATCHES/1 1  -40.0 -2.625  40.0 -2.625
$$HATCHES/1 1   40.0 -1.750 -40.0 -1.750
$$HATCHES/1 1  -40.0 -0.875  40.0 -0.875
$$HATCHES/1 1   40.0  0.000 -40.0  0.000
$$HATCHES/1 1  -40.0  0.875  40.0  0.875
$$HATCHES/1 1   40.0  1.750 -40.0  1.750
$$HATCHES/1 1  -40.0  2.625  40.0  2.625
```

Appendix C

Python code

C.1 Subroutine 1: Data file creator

C.1.1 One parameter data file creator

```
import os
import sys
import fileinput

#This subroutine opens the file called 'model' and look for
#the keyword 'busca' replacing its value by the 'parlist'
#value. It creates one file for each value of the list.

#DEFINE THE PARAMETER AND ITS LIST OF VALUES:

parname = 'ALPHA'
parlist = [1e-05, 1.3e-05, 1.711e-05]

#NAME OF THE MODEL FILE AND KEYWORD TO BE REPLACED:

model = 'model.dat'
busca = 'VALUE1'


#-----------------------------------------------------------#
#SUBROUTINE BODY (DO NOT CHANGE)
for param in parlist:
   name = model[:-4] + parname + 'v' + str(param)+ 'v' +
      '.dat'
   replacement = str(param)
   f = open(model, 'r+')
   f1 = open(name,'w')
   changes = 0
   linesChange = []
   for line in fileinput.input(model):
      if busca in line:
         changes = changes + 1
         f1.write(line.replace(busca,replacement))
         linesChange.append(fileinput.filelineno())
      else:
         f1.write(line)

   print('The number of changes in ' + name + ' is ' +
      str(changes))
   print('The changes are in lines: ' + str(linesChange))
```

```
   f1.close()
   f.close()
```

C.1.2 Two parameter data file creator (no combination)

```
import os
import sys
import fileinput

#This subroutine opens the file called 'model' and look for
#the keywords 'busca1' and 'busca2' replacing its values
#by the 'parlist1' and 'parlist2' values. It creates
#one file for each pair of values of the lists sharing
#the same index.


#DEFINE THE PARAMETERS AND ITS LISTS OF VALUES:

parname1 = 'YOUNG'
parname2 = 'HOT'

parlist1 = [68.95e+9, 106.86e+9, 195.12e+9, 226.84e+9]
parlist2 = [59.95e+9, 77.22e+9, 146.17e+9, 125.49e+9]


#NAME OF THE MODEL FILE AND KEYWORD TO BE REPLACED:

model = 'model.dat'

busca1 = 'VALUE1'
busca2 = 'VALUE2'


#-----------------------------------------------------------#
#SUBROUTINE BODY (DO NOT CHANGE)
for param1 in parlist1:
   for param2 in parlist2:
      if parlist1.index(param1) == parlist2.index(param2):
         name = model[:-4] + parname1 + 'v' + str(param1) +
            'v' + parname2 + 'w' + str(param2)+ 'w' +
            '.dat'
         replacepar1 = str(param1)
         replacepar2 = str(param2)
         f = open(model, 'r+')
         f1 = open(name,'w')
         changes1 = 0
         changes2 = 0
         linesChange1 = []
         linesChange2 = []
         for line in fileinput.input(model):
            b1 = False
            b2 = False
            if busca1 in line:
               b1 = True
```

```
            if busca2 in line:
               b2 = True

            if b1 & b2:
                  changes1 = changes1 + 1
                  changes2 = changes2 + 1
                  line = line.replace(busca1,replacepar1)
                  line = line.replace(busca2,replacepar2)
                  f1.write(line)
                  linesChange1.append(fileinput.filelineno())
                  linesChange2.append(fileinput.filelineno())
            elif b1:
               changes1 = changes1 + 1
               line = line.replace(busca1,replacepar1)
               f1.write(line)
               linesChange1.append(fileinput.filelineno())
            elif b2:
               changes2 = changes2 + 1
               line = line.replace(busca2,replacepar2)
               f1.write(line)
               linesChange2.append(fileinput.filelineno())
            else:
               f1.write(line)
         print('The total number of changes in ' + name + '
            is ' + str(changes1 + changes2))
         print('The ' + str(changes1) + ' changes in the
            parameter ' + parname1 + ' are in lines: ' +
            str(linesChange1))
         print('The ' + str(changes2) + ' changes in the
            parameter ' + parname2 + ' are in lines: ' +
            str(linesChange2))
         f.close()
         f1.close()
```

C.1.3 Two parameter data file creator (combination)

```
import os
import sys
import fileinput

#This subroutine opens the file called 'model' and look for
#the keywords 'busca1' and 'busca2' replacing its values
#by the 'parlist1' and 'parlist2' values. It creates one
#file for each possible combination of any pair of
#values defined in the lists.



#DEFINE THE PARAMETERS AND ITS LISTS OF VALUES:

parname1 = 'KRATI'
parname2 = 'GRATI'
parlist1 = [0.5, 0.7, 0.9]
parlist2 = [0.1, 0.2, 0.3, 0.4]
```

```
#NAME OF THE MODEL FILE AND KEYWORD TO BE REPLACED:

model = 'model.dat'

busca1 = 'VALUE1'
busca2 = 'VALUE2'



#-----------------------------------------------------------#
#SUBROUTINE BODY (DO NOT CHANGE)
for param1 in parlist1:
   for param2 in parlist2:
      name = model[:-4] + parname1 + 'v' + str(param1) +
         'v' + parname2 + 'w' +str(param2) + 'w' + '.dat'
      replacepar1 = str(param1)
      replacepar2 = str(param2)
      f = open(model, 'r+')
      f1 = open(name,'w')
      changes1 = 0
      changes2 = 0
      linesChange1 = []
      linesChange2 = []
      for line in fileinput.input(model):
         b1 = False
         b2 = False
         if busca1 in line:
            b1 = True
         if busca2 in line:
            b2 = True

         if b1 & b2:
               changes1 = changes1 + 1
               changes2 = changes2 + 1
               line = line.replace(busca1,replacepar1)
               line = line.replace(busca2,replacepar2)
               f1.write(line)
               linesChange1.append(fileinput.filelineno())
               linesChange2.append(fileinput.filelineno())
         elif b1:
            changes1 = changes1 + 1
            line = line.replace(busca1,replacepar1)
            f1.write(line)
            linesChange1.append(fileinput.filelineno())
         elif b2:
            changes2 = changes2 + 1
            line = line.replace(busca2,replacepar2)
            f1.write(line)
            linesChange2.append(fileinput.filelineno())
         else:
            f1.write(line)
      print('The total number of changes in ' + name + ' is
         ' + str(changes1 + changes2))
```

```
      print('The ' + str(changes1) + ' changes in the
         parameter ' + parname1 + ' are in lines: ' +
         str(linesChange1))
      print('The ' + str(changes2) + ' changes in the
         parameter ' + parname2 + ' are in lines: ' +
         str(linesChange2))
      f.close()
      f1.close()
```

C.2 Subroutine 2: Executable list

```
import os
import sys
import fileinput

#This subroutine make a list of the .dat files in a folder
#and create the executable batch file calling the 'exe'
#executable.

#DEFINE EXECUTABLE NAME AND MODEL NAME:

exe = 'mycomet64'
model = 'model.dat'


#-------------------------------------------------------------#
#SUBROUTINE BODY (DO NOT CHANGE)
name = exe + 'File.cmd'
path = os.getcwd()
listatotal = os.listdir(path)
listadat = []


#Data file list (execute)

for file in listatotal:
   if '.dat' in file:
      file = file[:-4]
      listadat.append(file)
   else:
      continue


#Write the autoexecutable batch (.cmd)

batch = open(name,'w')

batch.write('@echo off\n')
batch.write('\n')
batch.write('cd ' + path + '\n')

for dat in listadat:
   if dat != model[:-4]:
      batch.write(exe + ' ' + dat + '\n')
```

```
batch.write('pause\n')
batch.write('exit')

batch.close()
```

C.3 Subroutine 3: Plot generator

C.3.1 General plot generator

This code plots all the figures with a certain extension (cur01, cur02...)

```
import os
import sys
import fileinput
import re

#This subroutine creates a .txt file directly deliverable
    to Gnuplot.
#You can define the gnu characteristic such as: legend
    name, sizes...
#The value given to the legend can be extracted from file
    name
#(v = VALUE1 and w = VALUE2).

#PLOT EXTENSIONS TO BE DRAWN: (plot all files with
    extension in extensionList)
extensionList = ['.cur01']

#LEGEND NAME AND EXTRACTED VALUE FROM FILE:
legendName = 'Density'
value = 'v' #Write 'v' if value1 and 'w' if value2

#GNUPLOT CHARACTERISTICS:

dataStyle = 1 # linespoints = 1 // lines = 2
pointSize = 2.5 #Size of the data points
xlabel = '"Time (s)"' #Name of the x-axis
xlabelFontSize = 20 #Define the size of your xlabel font
xlabelOffset = [0 , -2] #Define the offset of the xlabel
    (x, y)
ylabel = '"Temperature"' #Name of the y-axis
ylabelFontSize = 20 #Define the size of your ylabel font
ylabelOffset = [-5, 0] #Define the offset of the ylabel (x,
    y)
xticsFontSize = 16 #Define the size of the number xtics
yticsFontSize = 16 #Define the size of the number ytics
key = 2 #0 key off // 1 key on // 2 key on box
keyLocation = 0 #0 default // 1 top // 2 bottom
KeyFontSize = 22 #Define the size of your key font
titleExtractedFromName = 0 # 0 False // 1 True

#-----------------------------------------------------------#
#SUBROUTINE BODY (DO NOT CHANGE)
path = os.getcwd()
```

```
listatotal = os.listdir(path)
listacur = []

for extension in extensionList:
#Plotting list
   listacur = []

   for file in listatotal:
      if extension in file:
         listacur.append(file)
      else:
         continue


   #Write the plotting (gnuplot) .txt file

   nameplot = 'plot' + extension[1:] + '.txt'
   plot = open(nameplot,'w')

   plot.write('cd ' + "'" + path + "'" + '\n')

   if dataStyle == 1:
      plot.write('set style data linespoints\n')
   else:
      plot.write('set style data lines\n')

   plot.write('set xlabel ' + xlabel + '\n')
   plot.write('set ylabel ' + ylabel + '\n')
   plot.write('set xlabel offset ' + str(xlabelOffset[0]) +
      ',' + str(xlabelOffset[1]) + '\n')
   plot.write('set ylabel offset ' + str(ylabelOffset[0]) +
      ',' + str(ylabelOffset[1]) + '\n')

   if key == 0:
      plot.write('unset key\n')
   elif key == 1:
      plot.write('set key\n')
   else:
      plot.write('set key box\n')

   if keyLocation == 1:
      plot.write('set key t\n')
   elif keyLocation == 2:
      plot.write('set key b\n')
   else:
      pass

   plot.write('set key font "Times-Roman, ' +
      str(KeyFontSize) + '\n')
   plot.write('set xlabel font "Times-Roman, ' +
      str(xlabelFontSize) + '\n')
   plot.write('set ylabel font "Times-Roman, ' +
      str(ylabelFontSize) + '\n')
   plot.write('set xtics font "Times-Roman, ' +
      str(xticsFontSize) + '\n')
   plot.write('set ytics font "Times-Roman, ' +
      str(yticsFontSize) + '\n')
```

```
   if dataStyle == 1:
      plot.write('set pointsize ' + str(pointSize) + '\n')
   plot.write('plot ')

   for cur in listacur:
      plot.write("'" + cur + "'")

      if titleExtractedFromName:
         splitList = cur.split(value)
         plot.write(" title " + "'" + legendName + " = " +
            splitList[1] + "'")
      if cur is not listacur[-1]:
         plot.write(',')

   plot.close()
```

C.3.2 Selective plot generator

This code plots the figures which match certain regular expression pattern.

```
import os
import sys
import fileinput
import re

#This subroutine create the .txt file to send to gnuplot in
    a selective way
#governed by a regular expression pattern.

#Define the pattern (regular expression) of plotting files:
#For information on RE visit:
    https://docs.python.org/2/library/re.html

plotName = 'plotS.txt' #Name of the plot
patternName = '.*' #RE pattern


#GNUPLOT CHARACTERISTICS:

dataStyle = 1 # linespoints = 1 // lines = 2
pointSize = 2.5 #Size of the data points
xlabel = '"Time (s)"' #Name of the x-axis
xlabelFontSize = 20 #Define the size of your xlabel font
xlabelOffset = [0 , -2] #Define the offset of the xlabel
    (x, y)
ylabel = '"Temperature"' #Name of the y-axis
ylabelFontSize = 20 #Define the size of your ylabel font
ylabelOffset = [-5, 0] #Define the offset of the ylabel (x,
    y)
xticsFontSize = 16 #Define the size of the number xtics
yticsFontSize = 16 #Define the size of the number ytics
key = 2 #0 key off // 1 key on // 2 key on box
keyLocation = 0 #0 default // 1 top // 2 bottom
KeyFontSize = 22 #Define the size of your key font
```

```
titleExtractedFromName = 0 # 0 False // 1 True


#-------------------------------------------------------------#
#SUBROUTINE BODY (DO NOT CHANGE)

path = os.getcwd()
listatotal = os.listdir(path)
lista = []
#Plotting list

for file in listatotal:
   if re.match(patternName, file):
      lista.append(file)
   else:
      continue

#Write the plotting (gnuplot) .txt file

plot = open(plotName,'w')

plot.write('cd ' + "'" + path + "'" + '\n')

if dataStyle == 1:
   plot.write('set style data linespoints\n')
else:
   plot.write('set style data lines\n')

plot.write('set xlabel ' + xlabel + '\n')
plot.write('set ylabel ' + ylabel + '\n')
plot.write('set xlabel offset ' + str(xlabelOffset[0]) +
    ',' + str(xlabelOffset[1]) + '\n')
plot.write('set ylabel offset ' + str(ylabelOffset[0]) +
    ',' + str(ylabelOffset[1]) + '\n')

if key == 0:
   plot.write('unset key\n')
elif key == 1:
   plot.write('set key\n')
else:
   plot.write('set key box\n')

if keyLocation == 1:
   plot.write('set key t\n')
elif keyLocation == 2:
   plot.write('set key b\n')
else:
   pass

plot.write('set key font "Times-Roman, ' + str(KeyFontSize)
    + '\n')
plot.write('set xlabel font "Times-Roman, ' +
    str(xlabelFontSize) + '\n')
plot.write('set ylabel font "Times-Roman, ' +
    str(ylabelFontSize) + '\n')
plot.write('set xtics font "Times-Roman, ' +
    str(xticsFontSize) + '\n')
```

```
plot.write('set ytics font "Times-Roman, ' +
    str(yticsFontSize) + '\n')

if dataStyle == 1:
   plot.write('set pointsize ' + str(pointSize) + '\n')
plot.write('plot ')

for cur in lista:
   plot.write("'" + cur + "'")

   if titleExtractedFromName:
      splitList = cur.split(value)
      plot.write(" title " + "'" + legendName + " = " +
         splitList[1] + "'")
   if cur is not lista[-1]:
      plot.write(',')

plot.close()
```